Adapting to the Long Tail in Language Understanding

Aakanksha Naik

CMU-LTI-22-002

Language Technologies Institute
School of Computer Science
Carnegie Mellon University
5000 Forbes Ave., Pittsburgh, PA 15213
www.lti.cs.cmu.edu

Thesis Committee:

Carolyn Rosé (Chair)
Jill Fain Lehman
Emma Strubell
Matt Gormley
Luke Zettlemoyer, University of Washington
and Meta AI Research

Submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy In Language and Information Technologies

Abstract

Advances in deep learning, especially self-supervised representation learning, have produced models that reach human parity on many benchmark datasets, which cover a variety of natural language understanding tasks. However, benchmark datasets are constructed from naturally occurring text, and are no exception to Zipf's law, containing a small proportion of highly frequent cases and a *long tail* of less frequent cases. Benchmark-driven evaluation and model development favors NLU models that perform well on the head, sidelining domains and phenomena that are underrepresented.

In this thesis, we adopt a two level conceptualization of the long tail: (i) macro-level according to broad dimensions of linguistic variation such as language, genre, topic, etc., and (ii) micro-level according to the presence or absence of specific linguistic phenomena such as numeracy, deixis, etc. With this conceptualization in mind, we focus on addressing three research questions about the applicability of domain adaptation to the long tail: (i) how can we best adapt between macro-level dimensions?, (ii) how can we best handle micro-level phenomena?, (iii) how do we evaluate performance on the long tail?

For adaptation at the macro level (low-resource domains), we propose: (i) likelihood-based instance weighting, an unsupervised adaptation technique that uses language model likelihoods to estimate source-target similarity, and (ii) domain-aware query sampling, an embedding similarity-based criterion to improve data efficiency during active learning. For micro-level adaptation (low-resource phenomena), we present an integrated architecture that incorporates knowledge/rules represented as ILP constraints into neural model training using a structured SVM framework. Finally, for long tail evaluation, we develop an evaluation paradigm called "stress tests", which allows us to identify micro long tail phenomena that models fail on by supplementing benchmark evaluation with evaluation on non-identically distributed phenomenon-focused test-only datasets.

Through a series of systematically designed case studies, we analyze and contrast the performance of these proposed techniques with existing transfer learning methods on information extraction and text classification tasks. Our goal is to identify promising categories of methods for the long tail, while mapping out their limits. This thesis takes preliminary steps towards aggregating a series of best practices that can facilitate informed selection from an arsenal of strong transfer methods, given a new long tail setting.

To my family

Acknowledgments

As a fledgling researcher embarking on the arduous journey of getting a PhD, the idea of reaching this point often felt like a pipe dream. Like many others, my journey was not entirely smooth sailing, but at some point, I discovered a way to cope with moments of deep frustration: reading acknowledgement sections from other people's theses. Seeing others list their pillars of support would remind me of all the people in my life who have given me their unwavering support, and strengthen my resolve to see this endeavour through, so that I could make them proud and one day, write a similar acknowledgements section dedicated to them. Now that I have (finally!) finished writing this thesis and am trying to convey my gratitude to those who made this possible, I find myself overwhelmed, and at a complete loss for words. Nevertheless, here is my attempt at appreciating my advisors, mentors, colleagues, family and friends, to whom I am deeply indebted for their encouragement.

First and foremost, I would like to thank my advisor, Carolyn Rosé, for her advice and support at every single step during this journey. I am incredibly grateful that she chose to take me on despite my inexperience and patiently guided me through the process of developing my research agenda and interests. I have learned so much from her over the years that I cannot fit everything into a small paragraph, but I hope to keep incorporating these lessons into my future research. She has also been a constant source of encouragement for me through fellowship applications, paper rejections and internship and job searches, counteracting my usual pessimism. Her support, over the last few years, has not solely been limited to research, and I am particularly grateful for her empathy and understanding during times when I had to deal with personal struggles.

I am very thankful to my committee members Jill Lehman, Emma Strubell, Matt Gormley and Luke Zettlemoyer for their invaluable support and feedback on my thesis research. I am especially indebted to Jill for going over this document, and many of my papers, with a fine toothcomb and bringing up interesting insights and additional analyses that could be pursued, while also identifying writing issues that slipped under my radar. I am extremely happy that I had the opportunity to work with her during my time at CMU because she has taught me so much about cognitive science, the history of AI, and most importantly, about being a woman in computer science. I also want to thank Emma for her suggestions during my thesis proposal, without which I do not think I would have undertaken the meta-analysis project that ended up providing such a cohesive framework for my thesis and being my favourite project out of my PhD. I thoroughly enjoyed meeting with all of you and getting your perspectives as I was working on this thesis, and I hope we can work together sometime in the future too.

Much of the work in this thesis would not have been possible without the support

of other faculty members at the Language Technologies Institute, my collaborators (both internal and external) and members of the TELEDIA lab. I want to thank Graham Neubig for his enthusiastic support and mentorship, which helped mold what started as a class project into my first conference paper. This was an instrumental experience for me and taught me a lot about research writing. I am very thankful to Eric Nyberg and Alan Black for their guidance when we were participating in the BioASQ challenge and Alexa challenge in 2017 as largely inexperienced students, and for their support during the PhD application process, without which I may not have been here. I would also like to thank Eduard Hovy for his invaluable feedback on many projects, and for amplifying my interest in computational semantics.

I am extremely grateful to my research soulmate Abhilasha Ravichander, without whom the stress tests and numeracy projects might not have seen the light of day. Our long research and thesis framing conversations have been a great source of inspiration for me throughout, and your feedback on early drafts of my projects (and sometimes emails) has been crucial for me. I also want to thank Khyathi Chandu and Aditya Chandrasekar, my co-conspirators during BioASQ and most of my masters - I do not think I would have gotten through the never-ending stream of assignments and course projects without your help and companionship.

I am heavily indebted to members of the TELEDIA group, both past and present, for being a great source of early feedback on projects, papers, and presentations, and I would like to particularly thank Hyeju Jang, Yohan Jo, Michael Yoder, Luke Breitfeller, Xinru Yan, Chris Bogart, and Shivani Poddar for contributing to or helping out with work included in this thesis. I want to give a special shout-out to Chas Murray for painstakingly looking after our infrastructure needs, none of the experiments in this work would have been possible otherwise! I also want to thank Stacey Young, Kate Schaich, and Mary Jo Bensasi for tirelessly ensuring that we do not miss any important administrative deadlines.

In addition to the TELEDIA lab, I want to extend my thanks to the members of the Epidemiology and Biostatistics section at the National Institutes of Health Clinical Center: Elizabeth Rasch, Julia Porcino, Denis Newman-Griffis, Chunxiao Zhou, Bart Desmet, Ayah Zirikly, Guy Divita, Hao-Ren Yao, Maryanne Sacco, Pei-Shu Ho, Jona Maldonado, Cricket Coale, Rafael Jimenez, Leslie Grubbs-King, Kaushik Gedela, Josh Chang, and Alex Marr. All of you provided me immense support during the crucial years of my PhD, and I could not have asked for more understanding and helpful colleagues.

Many collaborators outside the LTI and the NIH have also been great sources of feedback and inspiration over the years. I am very thankful to James Antaki and Lisa Lohmueller who provided me my first real introduction to interdisciplinary research during my early years at CMU. I am very grateful to Pararth Shah, Shane Moon, Bing

Liu, and Honglei Liu for their mentorship during my internship at Facebook, which was my first experience with NLP research in industry and led me to re-calibrate my aims. I am also very indebted to Tom Hope, Lucy Lu Wang, and Sergey Feldman for their brilliant, attentive and empathetic mentorship during my internship at the Allen Institute for Artificial Intelligence. The stimulating and encouraging environment you created played a significant role in making returning to AI2 as a full-time researcher a dream goal for me.

My research journey in NLP started during the summer of my junior year in undergrad, and I want to thank Anton Leuski for taking me on as an intern and introducing me to the world of open research problems in this field. I also want to extend my gratitute to Partha Talukdar, my undergraduate thesis supervisor. The time I spent working with him at the MALL lab only cemented my desire to pursue a PhD. I also want to thank some of my teachers from high school, Rema mam, Amba mam, Rekha mam, Rathi sir, Kamal sir, and Nishant sir, who always encouraged my academic ambitions and taught me to push myself out of my comfort zone.

I am extremely grateful for all my friends who made this journey happier and much more fun than it would have been otherwise. Thanks to my ex-officemates Chan Young Park and Anjalie Field, I wish we would have had more time to have meandering conversations before COVID happened. I am also thankful to Khyathi, Abhilasha, Aditya, Shruti Palaskar, Danish Pruthi, Mansi Gupta, Siddharth Dalmia, and Shruti Rijhwani for fun conversations, dinners and walks that provided much-needed breaks. I am grateful to Anusha Bagalkotkar and Rucha Vaidya, I could not have asked for better roommates during my first year living in the US. I am very thankful for the constant love and support from Mohit Gupta, Rucha Panchabhai, Srabasti Nandi, Prajakta Joshi, and Anshita Srivastava - you kept me sane and reminded me that I had a life outside of work (and I'm looking forward to our annual reunions)!

Finally, I want to thank my family: Mahabaleshwar ajoba, Malini aaji, Sonali kaku, Tushar kaka, Jaya kaku, Medha maushi, Vijay kaka, Varsha maushi, Anil kaka, Anuradha, Ajinkya, Ananth, Mayuri didi, Shreyas jijaji, Vrushali didi, Sayali didi, Shrikant jijaji, Shreyas dada, and Savya. I also want to thank Vidyadhar ajoba, Vinaya aaji, Ajay mama, Vaishali mami, and Deepak kaka - I wish I could have celebrated this achievement with you. Last, but not the least, I want to thank my mom, dad, and my brother Atharva for putting up with my crankiness and frustration that progressively worsened as I got closer to the finish line. Thank you mom and dad for cultivating a love for science in me, for unconsciously nourishing that love by buying me lots of books about scientists and inventors, and for supporting me through this journey even though it may not have been what you envisioned me doing. Thanks mom for having bigger dreams for me than I did for myself, and thanks dad for proofreading my thesis despite finding it "dry" - now I can claim that more than one person did read my thesis!

Contents

List of Figures

List of Tables

CHAPTER 1

Introduction

Enabling machines to achieve *human-like* competence at understanding the intricacies and ambiguities of natural language has been a longstanding goal in artificial intelligence. Language understanding is an important sub-goal that machines must accomplish to pass classic tests designed to measure intelligent behavior, such as the Turing Test, proposed in 1950, and the Winograd Schema Challenge (Levesque et al., 2012). Early attempts to build natural language understanding systems relied heavily on keyword matching and heuristic rules. One of the earliest attempts culminated in the development of STUDENT (Bobrow, 1964), a system that could solve algebra word problems expressed using a restricted set of English. Subsequent years saw the development of ELIZA (Weizenbaum, 1966), a rule-based chatbot that could carry out conversations on many topics, and SHRDLU (Winograd, 1972), a system that could understand simple English sentences in a restricted world of children's blocks to guide movements. While these systems demonstrated the feasibility and utility of endowing machines with the ability to understand natural language, they were not broad-coverage and were largely restricted to specific closed worlds or domains.

Over the years, the increasing availability of data coupled with the restrictive nature of rule-based systems drove a revival of empiricism in language understanding (Church, 2011). This wave of empiricism relied heavily on a benchmark-driven approach to natural language understanding, a protocol dating back to the construction of the Penn TreeBank (Marcus et al., 1993). As described in this pioneering work, the key principle behind benchmark-driven NLU is the idea that rapid progress can be made if we investigate and learn to model those phenomena that occur *most centrally* (or frequently) in free text. Developing benchmarks allows for more principled and controlled comparison of modeling advancements by providing a level playing field for testing new approaches. Since then, a slew of benchmark datasets have been developed for several NLU tasks such as question answering (Rajpurkar et al., 2016), natural language inference (Bowman et al.,

2015; Williams et al., 2018), dialog state tracking (Budzianowski et al., 2018), etc. Recent work has taken this a step further and developed leaderboards that allow models to be tested on a wide range of NLU tasks and rank them based on their aggregate performance across tasks (McCann et al., 2018; Wang et al., 2019c,b).

In tandem with work on benchmark development, research on distributional semantic models, which use vector spaces to represent words, made tremendous strides, moving from count-based DSMs (Landauer and Dumais, 1997; Schütze, 1998) to semi-supervised neural models pre-trained on language processing tasks (Collobert and Weston, 2008). These advancements most recently culminated in the development of large-scale neural language models pre-trained on massive amounts of data (Peters et al., 2018; Devlin et al., 2019; Yang et al., 2019; Liu et al., 2019b), which subsequently serve as a source of semantic vector representations. Unlike old-school symbolic NLU systems, these models are touted to be highly broad-coverage, and excel on a wide range of benchmarks and leaderboards with minimal modifications, even achieving human parity on some (Wang et al., 2019c). Pre-trained language models have become a de-facto starting point for building NLU models, due to their excellent performance.

However, an important caveat is that these models have primarily been tested on benchmarks and leaderboards, which are only *samples* from all available text, and therefore not equally representative of all domains or linguistic phenomena. Drawing an analogy to Zipf's law, benchmarks and leaderboards are dominated by a small proportion of high-frequency common cases, and leave out a *long tail* of low-frequency cases in NLU. Consequently, models trained and evaluated on benchmarks can achieve high performance by optimizing for high-frequency cases, without developing mechanisms to handle long tail cases. This raises a natural question: how well do benchmark-trained models perform on the long tail, and can we develop methods to *adapt* them to the long tail better and *evaluate* them more comprehensively? Notably, adapting benchmark-trained models to the long tail can be also viewed from the perspective of tackling distributional shift, a problem with a rich history of study in the field of transfer learning.

1.1 Conceptualizing the Long Tail in Language Understanding

This thesis adopts a two level conceptualization of the long tail: (i) macro-level, and (ii) micro-level. At the macro-level, the space of all available text is categorized according to broad dimensions of linguistic variation such as language, genre, topic, register, etc., and texts belonging to dimensions that are underrepresented in benchmarks and leaderboards constitute the macro long tail. For example, most standard benchmarks do not contain text from high-expertise domains that require specialized knowledge to understand them, such as biomedical text, clinical text, financial text, etc. Therefore, such domains can be considered to be part of the long tail. Due to the constantly evolving nature of language, there isn't a strong consensus on what constitutes a comprehensive, or exhaustive set of dimensions of linguistic variation (Plank, 2016), despite prior work on identifying such dimensions via corpus linguistics (Biber, 1991). We do not tackle this question, but instead

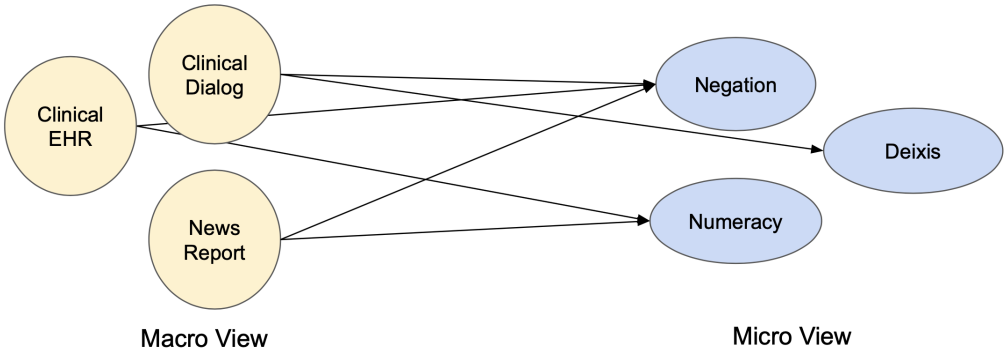


Figure 1.1: Two different views of the long tail in natural language understanding. Note that categorization at both levels is multi-dimensional, i.e., a piece of text may contain multiple types of linguistic variation or linguistic phenomena.

adopt a commonly used set of existing dimensions that suits the scope of our work, and our focus on NLU from text (further discussed in Chapter 2). At the micro-level, the space of all available text is categorized according to the presence of specific linguistic phenomena such as negation, deixis, etc., which must be tackled to understand the meaning conveyed. Again, examples containing phenomena that are infrequent in standard benchmark datasets make up the micro long tail. For example, the MultiNLI dataset (Williams et al., 2018), which is a benchmark dataset for natural language inference and sentence understanding, contains very few examples involving numerical reasoning (Naik et al., 2018). Hence examples requiring numeracy can be considered to be part of the micro long tail. As with macro-level dimensions, there is no static, centrally agreed-upon exhaustive list of linguistic phenomena. Additionally, most datasets do not annotate their examples for the presence (or absence) of linguistic phenomena, which adds an additional layer of difficulty to micro-level categorization.

Figure 1.1 presents a pictorial representation of this two-level categorization of the long tail in NLU. Since the proportions of various linguistic phenomena differ across macro dimensions such as languages, genres, etc., the macro and micro level long tails are intricately inter-connected. For example, while uncommon in general, negation is highly common in clinical records, with nearly 50% of mentioned conditions (or entities) being negated according to Chapman et al. (2001). This interconnection is highly valuable, as it allows us to use the macro-level long tail as a lens to study the micro-level long tail, a strategy we use in Chapters 2 and 3.

1.2 Research Questions

This thesis focuses on studying and addressing three key questions arising from the existence of the long tail in NLU:

1. **How can we best adapt benchmark-trained models across macro long tail dimensions?**

This is an important question to study since several real-world applications of NLU technologies involve macro long tail domains (e.g., clinical and financial applications), and benchmark advancements don't always transfer equally to such domains. To answer this, we explore the applicability of existing transfer learning methods, while simultaneously developing new transfer methods motivated by our settings of interest. Through case studies on multiple macro long tail domains that differ along various linguistic dimensions, we try to develop a better understanding of which categories of adaptation methods work best on different kinds of variation.

2. **How can we best equip benchmark-trained models to handle micro long tail phenomena?** This question is also crucial since, as discussed earlier, certain linguistic phenomena may be underrepresented in benchmarks, but be much more prominent or central in other tasks/domains. Additionally, even if a phenomenon is extremely infrequent, learning to model it may be an interesting linguistic question. To tackle this question, we conduct a case study exploring applicability of both existing and newly proposed transfer methods in a setting in which macro dimensions are held constant. This ensures that varying proportions of linguistic phenomena are primarily responsible for underlying differences.
3. **How can we comprehensively evaluate model performance on the long tail?** This is a key problem because benchmark evaluation under-emphasizes infrequent examples. This results in overly optimistic estimates of model performance, which are not reflected when the same models are applied to macro/micro long tail phenomena. We propose a new evaluation paradigm called stress testing that offers more comprehensive and realistic estimates of model performance. Rather than replacing benchmark-based evaluation, we recommend stress testing as a supplementary evaluation mechanism, to be adopted in addition to benchmark evaluation. Through multiple case studies, we demonstrate the utility of this paradigm in identifying micro long tail phenomena that benchmark-trained models are unable to handle.

1.3 Thesis Overview

Thesis Statement: *Through an extensive series of case studies, we identify promising categories of transfer learning methods for adaptation to the long tail, while mapping out their limits. Our work takes preliminary steps towards aggregating a series of best practices, facilitating informed selection from an arsenal of strong transfer methods, when presented with a new setting.*

The rest of this thesis is organized as a series of case studies designed to build up this set of best practices, while addressing our key research questions. Though we describe the motivations behind specific experimental setups for each case study in detail in the corresponding chapter, our study designs are broadly motivated by three principles:

- Ensuring inclusion of strong transfer methods from the most promising categories identified (so far).

- Moving beyond overall performance, and using extensive qualitative and quantitative analyses to better understand model strengths and weaknesses.
- Exploring negative results comprehensively as a way to map out performance limits of various methods.

We briefly summarize how our case studies tackle each research question below:

Addressing RQ1: Chapters 2 and 3 focus on understanding the performance of both existing and newly proposed transfer learning methods in the context of adaptation to macro long tail domains. Chapter 2 begins this journey by providing a birds-eye view of transfer learning research, including a hierarchical taxonomy of adaptation method, via a qualitative meta-analysis of representative papers in the field, and identifying research gaps that must be added for improved macro adaptation. We then address one of these gaps (studying adaptation under data-scarce settings) through an extensive case study on sequence labeling tasks such as entity and event extraction from clinical narratives. Our results, supported by extensive quantitative analyses and qualitative error analyses, indicate the promise of loss augmentation and pseudo-labeling methods, especially in an unsupervised adaptation setting. Part of the work presented in this chapter (Naik et al., 2021a) is set to appear in TACL 2022. Chapter 3 builds further on this work in two ways: (i) bringing two additional domains under our purview, of which one is a non-narrative domain, and (ii) proposing new adaptation methods to advance the development of certain under-researched method categories. We create two new event extraction test sets for the domains of clinical narratives and clinical conversations to facilitate domain expansion. The two new adaptation methods we propose include: (i) an unsupervised instance weighting method that leverages language model likelihoods for source-target similarity, and (ii) a domain-aware query sampling criterion for active learning methods that leverages source-target distance to improve data-efficiency in a limited supervision setting. We conduct two case studies to test these new methods, in addition to strong existing baselines on our expanded set of domains. Our studies primarily focus on the task of event extraction, though we conduct additional experiments with entity extraction in the active learning study for broader understanding. These studies further support the superiority of loss augmentation methods for narrative domains, indicate that pretraining methods might be stronger contenders for non-narrative domains, and demonstrate negligible benefits of the active learning method category in the limited supervision setting. Parts of this chapter were published as Naik and Rosé (2020) at ACL 2020, and Naik et al. (2021b) at EACL 2021.

Addressing RQ2: Chapter 4 studies the performance of benchmark-trained models on the micro long tail through the lens of a case study on temporal ordering of long-distance event pairs. Since we want underlying differences to primarily stem from varying distributions of linguistic phenomena, we design a setting in which all macro dimensions are held constant. We achieve this by developing a new dataset for the task of ordering long-distance event pairs by augmenting an existing temporal ordering dataset that is primarily focused on short-distance temporal ordering. Since both datasets comprise of the same set of documents, we have a high degree of macro dimension consistency. We explore the performance of loss augmentation methods from the model-centric category and

pseudo-labeling methods from the data-centric category, since these two categories emerged as strong contenders for macro-level adaptation. The loss augmentation method tested is a newly proposed joint BiLSTM+ILP model architecture that allows incorporation of predefined task-specific heuristics (e.g., transitivity) into the loss function during model training. Our case study demonstrates that both methods provide performance boosts (with pseudo-labeling being stronger), but combining both methods has largely negative results. Parts of this chapter were originally published as Naik et al. (2019) at SIGDIAL 2019 and Breitfeller et al. (2021) on arxiv.

Addressing RQ3: To address the question of comprehensive evaluation on the long tail, Chapter 5 proposes an evaluation paradigm called *stress tests*. This paradigm is primarily motivated by the observation that following traditional identically distributed evaluation paradigms results in test sets that sideline the same set of micro long tail phenomena as training sets. Hence, we propose the use of stress tests, which are defined as phenomenon-focused non-identically distributed test-only datasets used to supplement traditional benchmark evaluation. In addition to more stringent evaluation, we hope that focus on a single linguistic phenomenon (or a restricted subset of related phenomena) can help uncover actionable insights about model ability to deal with micro long tail phenomena. We carry out two case studies to demonstrate the utility of stress test-based evaluation. Our first case study builds a stress test-based evaluation platform for natural language inference, while the second case study builds a stress test-based evaluation platform for quantitative reasoning in natural language inference. We demonstrate the effectiveness of these evaluation platforms in isolating micro long tail phenomena that state-of-the-art models fail to perform well on, despite demonstrating high performance on the NLI task. Parts of this chapter were originally published as Naik et al. (2018) at COLING 2018, and Ravichander et al. (2019) at CoNLL 2019.[1]

Finally, Chapter 6 summarizes our conclusions from these case studies, the contributions and limitations of this thesis, and highlights directions for future work.

[1]First-authorship for both papers is shared jointly with Abhilasha Ravichander

CHAPTER 2

Constructing a Systematic View of the Long Tail

Natural language understanding (NLU) has made massive progress driven by large benchmarks, paired with research on transfer learning to broaden its impact. But benchmarks are dominated by a small set of frequent phenomena, leaving a long tail of infrequent phenomena underrepresented. This chapter begins developing a systematic view of the long tail and reflects on the question: *have transfer learning methods sufficiently addressed performance of benchmark-trained models on the long tail?* Since benchmarks do not list included/excluded phenomena, we conceptualize the long tail using macro-level dimensions such as underrepresented genres, topics, etc. We assess trends in transfer learning research through a qualitative meta-analysis of 100 representative papers on transfer learning for NLU. Our analysis asks three questions: (i) Which long tail dimensions do transfer learning studies target? (ii) Which properties help adaptation methods improve performance on the long tail? (iii) Which methodological gaps have greatest negative impact on long tail performance? Our answers to these questions highlight major research avenues in transfer learning for the long tail. Finally, we perform a case study comparing the performance of various adaptation methods on clinical narratives to show how systematically conducted meta-experiments can provide insights that enable us to make progress along these future avenues.

2.1 Introduction

"There is a growing consensus that significant, rapid progress can be made in both text understanding and spoken language understanding by investigating those phenomena that occur most centrally in naturally occurring unconstrained materials and by attempting to automatically extract information

about language from very large corpora." (Marcus et al., 1993)

Since the construction of the Penn Treebank, using annotated IID (independent and identically distributed) benchmark datasets to measure and drive progress in model development has been a central tenet in natural language processing. Benchmark datasets have been developed for several core NLP tasks such as part-of-speech tagging (Marcus et al., 1993), syntactic parsing (Taylor et al., 2003), named entity recognition (Tjong Kim Sang and De Meulder, 2003), dependency parsing (Buchholz and Marsi, 2006), etc. Early efforts in development of benchmark datasets primarily relied on sourcing annotations from linguists. However, the advent of crowdsourcing made it feasible to collect larger benchmark datasets while reducing expert annotation effort, since annotations could now be obtained from laypeople. In recent years, crowdsourcing has contributed to the creation of several large-scale "landmark" datasets for key tasks in natural language understanding such as question answering (Rajpurkar et al., 2016), natural language inference (Bowman et al., 2015), commonsense reasoning (Talmor et al., 2019), etc.

Benchmark datasets are an excellent method of precisely evaluating the incremental impact of modeling advancements since they offer a controlled testbed with minimum variance. This realization has birthed the trend of using "leaderboards" to track progress in natural language understanding. Leaderboards are constructed by collecting several benchmark datasets focused on diverse natural language understanding tasks (e.g., sentiment analysis, textual similarity, natural language inference etc.). These datasets may optionally be recast to follow a consistent task formalization. For example, all tasks may be recast in such a way that they require a model to take a sentence pair as input and produce an output label (label spaces differ across tasks). These leaderboards do away with most variance that may arise from different dataset choices and task formalizations, and provide a birds-eye view of key modeling advancements that have produced large gains across a multitude of NLU tasks. The first attempt to construct such a leaderboard was made by McCann et al. (2018), who constructed the DecaNLP leaderboard by recasting 10 tasks into a question answering format. DecaNLP was followed by the construction of the GLUE (Wang et al., 2019c) and SuperGLUE (Wang et al., 2019b) benchmark leaderboards, which are now central to evaluation of NLU models.

Ideally, to provide maximum utility, leaderboards and shared benchmark corpora must be diverse and comprehensive, which can be addressed at both levels of the long tail: (i) macro-level dimensions such as language, genre, topic, etc., and (ii) micro-level dimensions such as specific language phenomena. However, diversity and comprehensiveness is not straightforward to achieve.

According to Zipf's law, many micro-level language phenomena naturally occur infrequently and will be relegated to the long tail, except in cases of intentional over-sampling. Additionally, annotation issues such as comprehensibility to laypeople,[1] low annotator agreement, etc. also contribute towards the relegation of some micro-level phenomena to the long tail. At the macro level, various sampling issues such as the availability of raw texts, restrictions on data sharing, in

[1]Sometimes expert knowledge is required to comprehend and annotate a text (e.g., clinical notes, financial reports, literature, etc.). Building a benchmark from such data requires access to domain experts which can be hard to obtain.

addition to the advantages of restricting community focus to a specific set of benchmark corpora and limitations in resources, lead to portions of the macro-level space being under-explored. This can further cause certain micro-level phenomena to be under-represented. For example, since most popular coreference benchmarks focus on English narratives, they do not contain many instances of zero anaphora, a phenomenon quite common in other languages (e.g., Japanese, Chinese). In such situations, model performance on benchmark corpora may not be truly reflective of expected performance on micro-level long tail phenomena, raising questions about the ability of state-of-the-art models to generalize to the long tail.

Most benchmarks do not explicitly catalogue the list of micro-level language phenomena that are included or excluded in the sample, which makes it non-trivial to construct a list of long tail micro-level language phenomena. Hence, we use the alternate macro-level conceptualization of the long tail, i.e., undersampled portions of the macro-level space are treated as proxies for long tail micro-level phenomena. These undersampled *long tail* macro-level dimensions highlight gaps and present potential new challenging directions for the field. Therefore, periodically taking stock of research to identify long tail macro-level dimensions can help in highlighting opportunities for progress that have not yet been tackled. This idea has been gaining prominence recently; for example, Joshi et al. (2020) survey languages studied by NLP papers, providing statistical support for the existence of a macro-level long tail of low-resource languages.

In this chapter, our goal is to start constructing a systematic view of the long tail in transfer learning for NLU by characterizing the macro-level long tail and efforts that have tried to address it from transfer learning research. Large benchmarks have driven much of the recent methodological progress on NLU (Bowman et al., 2015; Rajpurkar et al., 2016; McCann et al., 2018; Talmor et al., 2019; Wang et al., 2019c,b), but the generalization abilities of benchmark-trained models to the long tail have been unclear. In tandem, the NLP community has been successfully developing transfer learning methods to improve generalization of models trained on NLU benchmarks (Ruder et al., 2019). The goal of transfer learning research is to tackle the macro-level long tail in NLU, leading to the question: *how far has transfer learning addressed performance of benchmark models on the NLU long tail, and where do we still fall behind?* Probing further, we perform a qualitative meta-analysis of a representative sample of 100 papers on transfer learning in NLU. We sample these papers based on citation counts and publication venues (§2.2.1), and document 7 facets for each paper such as tasks and domains studied, adaptation settings evaluated, etc. (§2.2.2). Adaptation methods proposed (or applied) are documented using a hierarchical categorization described in §2.2.3, which we develop by extending the hierarchy from Ramponi and Plank (2020). With this information, our analysis focuses on three questions:

- **Q1:** What long tail macro-level dimensions do transfer learning studies target? Here dimensions include tasks, domains, languages and adaptation settings covered in transfer learning research.
- **Q2:** Which properties help adaptation methods improve performance on long tail dimensions?

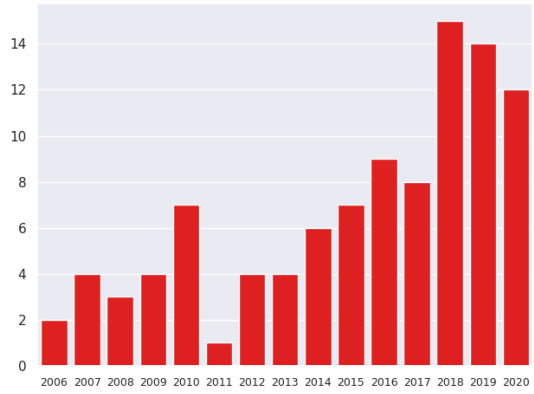


Figure 2.1: Distribution of meta-analysis sample papers across years.

- **Q3:** Which methodological gaps have greatest negative impact on long tail performance?

The rest of the chapter presents thorough answers to these questions, laying out avenues for future research on transfer learning that more effectively address the macro-level long tail in NLU. For Q1, based on statistics, we observe that transfer learning research has a tendency to sideline certain types of tasks, languages, domains, and adaptation settings. Thus though studies have attempted to evaluate on the long tail, coverage is far from comprehensive. For Q2, from studies that evaluate adaptation methods on long tail domains, we identify two useful properties, that have been sidelined in recent research: (i) incorporating source-target domain distance, and (ii) incorporating a nuanced view of domain variation instead of treating it as a dichotomy (source vs target). For Q3, we identify three major gaps: (i) combining adaptation methods, (ii) incorporating extra-linguistic knowledge (e.g., ontologies), and (iii) application to data-scarce adaptation settings such as unsupervised adaptation, online adaptation, etc. These gaps present avenues for future research in transfer learning for the long tail. We also present a case study to demonstrate that our meta-analysis framework can be used to systematically design and conduct experiments that provide insights that enable us to make progress along these avenues.

2.2 Meta-Analysis Framework

2.2.1 Sample Curation

For our meta-analysis, we gather a representative sample of work on transfer learning in NLU from the December 2020 dump of the Semantic Scholar Open Research Corpus (S2ORC) (Lo et al., 2020). First, we extract all papers published at nine prestigious *CL venues: ACL, EMNLP, NAACL, EACL, COLING, CoNLL, SemEval, TACL, and CL. This results in 25,141 papers, which are filtered to retain those containing the terms “domain adaptation” or “transfer learning” in the title or abstract, producing a set of 382 abstracts after duplicate removal. Search scope is limited to

Venue	#Papers	#TL Papers
Association for Computational Linguistics (ACL)	7200	149
Empirical Methods in Natural Language Processing (EMNLP)	4160	127
North American Chapter of the Association for Computational Linguistics (NAACL)	2943	52
European Chapter of the Association for Computational Linguistics (EACL)	1290	11
International Conference on Computational Linguistics (COLING)	4965	39
Conference on Natural Language Learning (CoNLL)	778	10
International Workshop on Semantic Evaluation (SemEval)	1632	33
Transactions of the Association for Computational Linguistics (TACL)	397	10
Computational Linguistics (CL)	1776	4

Table 2.1: Distribution of papers across venues in the complete corpus and the transfer learning subset.

title and abstract in order to prefer papers that focus on transfer learning rather than ones including a brief discussion or experiment on transfer learning as part of an investigation of something else. Table 2.1 shows the distribution of papers across venues in the complete corpus as well as the transfer learning subset.

We manually screen this subset and remove abstracts that are not eligible for our NLU-focused analysis (e.g., papers on generation-focused tasks like machine translation), leaving us with a set of 266 abstracts. From this, we construct a final meta-analysis sample of 100 abstracts via application of two inclusion criteria. Per the first criterion, all abstracts with 100 or more citations are included since they are likely to describe landmark advances. Then, remaining abstracts (to bring our meta-analysis sample to 100) are randomly chosen, after discarding ones with no citations.[2] The random sampling criterion ensures that we do not neglect studies that study less mainstream topics by focusing solely on highly-cited work. This produces a final representative sample of transfer learning work for our meta-analysis. Figure 2.2 gives an overview of our sample curation process via a PRISMA (Preferred Reporting Items for Systematic Reviews and Meta-Analyses) diagram (Page et al., 2021), while Figure 2.1 shows the year-wise distribution of the sample. Following is the complete list of all papers included in our final meta-analysis sample:

Papers with >= 100 citations: Blitzer et al. (2006), Daumé III (2007), Jiang and Zhai (2007), Blitzer et al. (2007), Chan and Ng (2007), Finkel and Manning (2009), McClosky et al. (2010), Chiticariu et al. (2010), Subramanya et al. (2010), Prettenhofer and Stein (2010), Li et al. (2012), Plank and Moschitti (2013), Eisenstein (2013), Liu et al. (2015), Nguyen and Grishman (2015), Zarrella and Marsh (2016), Søgaard and Goldberg (2016), Mou et al. (2016), Conneau et al. (2017),

[2] 23% of the papers from the final sample have 100 or more citations. The remaining randomly sampled papers have a mean citation count of 28.4, according to citation data from Semantic Scholar.

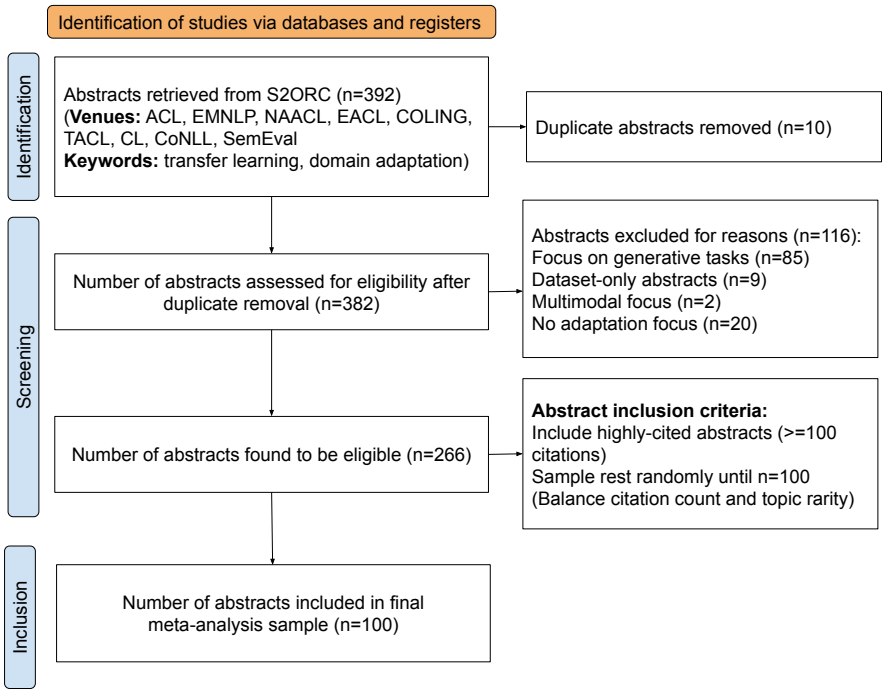


Figure 2.2: PRISMA diagram explaining our sample curation process.

Yang et al. (2017), Cer et al. (2018), Howard and Ruder (2018), Liu et al. (2019a)

Remaining papers: Chan and Ng (2006), Tsuboi et al. (2008), Arnold et al. (2008), Agirre and Lopez de Lacalle (2008), Jeong et al. (2009), Agirre and Lopez de Lacalle (2009), Tan and Cheng (2009), Umansky-Pesin et al. (2010), Rai et al. (2010), Chang et al. (2010), Yu and Kübler (2011), Szarvas et al. (2012), Dhillon et al. (2012), Mohit et al. (2012), Heilman and Madnani (2013), Scheible and Schütze (2013), Plank et al. (2014), Monroe et al. (2014), Jochim and Schütze (2014), Nguyen et al. (2014), Braud and Denis (2014), Passonneau et al. (2014), Yang and Eisenstein (2015), Yin et al. (2015), Ji et al. (2015), Yang et al. (2015), Al Boni et al. (2015), Kim et al. (2016), Abdelwahab and Elmaghraby (2016), Huang and Lin (2016), Sapkota et al. (2016), Gong et al. (2016), Pilán et al. (2016), Duong et al. (2017), Tourille et al. (2017), Zhang et al. (2017), Kim et al. (2017), Giménez-Pérez et al. (2017), Wu et al. (2017), Chen et al. (2018a), Hangya et al. (2018), Rodriguez et al. (2018), Xing et al. (2018), Wang et al. (2018), Lin and Lu (2018), Gee and Wang (2018), Alam et al. (2018), Romanov and Shivade (2018), Fares et al. (2018), Yang et al. (2018), Jiao et al. (2018), Huang et al. (2018), Vlad et al. (2019), Kamath et al. (2019), Li et al. (2019a), Chen and Qian (2019), Wiedemann et al. (2019), Beryozkin et al. (2019), Dereli and Saraclar (2019), Aggarwal and Sadana (2019), Li et al. (2019b), Huang et al. (2019), Johnson et al. (2019), Karunanayake et al. (2019), Wang et al. (2019a), Lison et al. (2020), Akdemir (2020), Chalkidis et al. (2020), Naik and Rose (2020), Lee et al. (2020), Tamkin et al. (2020), Chen et al. (2020b), Yan et al. (2020), Wright and Augenstein (2020), Vu et al. (2020), Schröder and Biemann

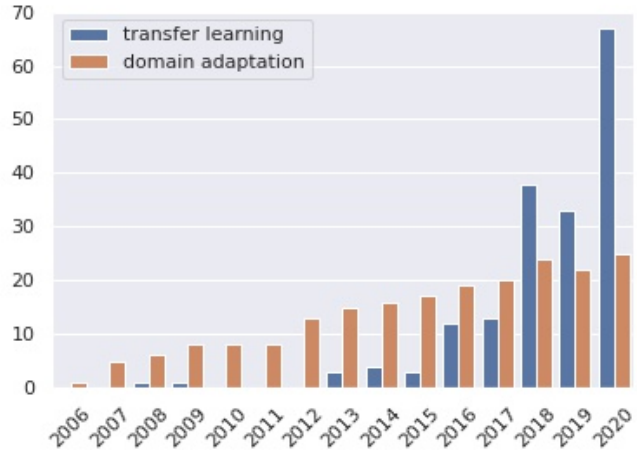


Figure 2.3: Distribution of papers retrieved by our search strategy across search terms and years.

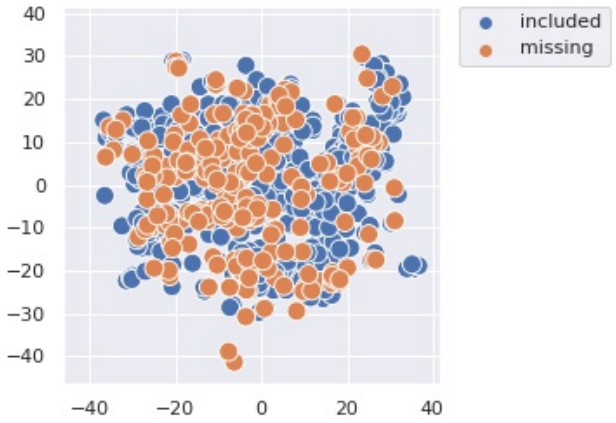


Figure 2.4: TSNE visualization of our meta-analysis sample alongside additional transfer learning papers missed by our keyword search.

(2020), Keung et al. (2020)

Characterizing limitations of our curation process: Since our sample curation process primarily relies on a keyword-based search, it might miss relevant work that does not use any of these keywords. To characterize the limitations of our curation process, we employ two additional strategies for relevant literature identification:

- **Citation graph retrieval:** Following Blodgett et al. (2020), we include all abstracts that cite or are cited by abstracts included in our keyword-retrieved set of 382 abstracts. This retrieves 3727 additional abstracts, but many of these works are cited for their description or introduction of new tasks, datasets, evaluation metrics, etc. Therefore, we discard all works that do not have the

Cat	Tasks Included
TC	Text classification tasks like sentiment analysis, hate speech detection, propaganda detection, etc.
NER	Semantic sequence labeling tasks like NER, event extraction, etc.
POS	Syntactic sequence labeling tasks like POS tagging, chunking, etc.
NLI	Natural language inference, NLU Tasks recast as NLI (e.g., GLUE)
SP	Structured prediction tasks such as entity and event coreference
WSD	Word sense disambiguation
TRN	Text ranking tasks (e.g., search)
TRG	Text regression tasks
RC	Reading comprehension
MF	Matrix factorization
LI	Lexicon induction
SLU	Spoken language understanding

Table 2.2: Categorization of tasks studied.

words "adaptation" or "transfer", leaving 282 new abstracts.

- **Nearest neighbor retrieval:** We use SPECTER (Cohan et al., 2020) to compute embeddings for all abstracts included in our keyword-retrieved set, as well as all abstracts in the ACL anthology. Then we retrieve the nearest neighbor for every abstract in our keyword-retrieved set, which results in the retrieval of 262 new abstracts.

Combining abstracts returned by both strategies, we are able to identify 510 additional works. However, while going over them manually, we notice that despite our noise reduction efforts, not all abstracts describe transfer learning work. We perform an additional manual screening step to discard such work, which leaves us with a final set of 232 additional papers. To identify whether the exclusion of these papers from the initial sample may have led to visible gaps or blind spots in our meta-analysis, we perform a TSNE visualization of SPECTER embeddings for both keyword-retrieved papers and this additional set of papers. Figure 2.4 presents the results of this visualization and indicates that there aren't visible distributional differences between the two subsets. Hence, though our sample curation strategy is imperfect, it seems unlikely that our final observations from the meta-analysis would be very different.

2.2.2 Meta-Analysis Facets

For every paper from our meta-analysis sample, we document the following key facets:

- **Task(s):** NLP task(s) studied in the work. Tasks are grouped into 12 categories based on task formalization and linguistic level (e.g., lexical, syntactic, etc.), as shown in Table 2.2.
- **Domain(s):** Source and target domains and/or languages studied, along with datasets used.
- **Task Model:** Base model used for the task, to which domain adaptation algorithms are applied.

- **Adaptation Method(s):** Domain adaptation method(s) proposed or used in the work. Adaptation methods are grouped according to the categorization shown in Figure 2.5 (details in §2.2.3).
- **Adaptation Baseline(s):** Baseline adaptation method(s) to compare new methods against.
- **Adaptation Settings:** Source-target transfer settings explored in the work (e.g., unsupervised adaptation, multi-source adaptation, etc.).
- **Result Summary:** Performance improvements (if any), performance differences across multiple source-target pairs or methods, etc.

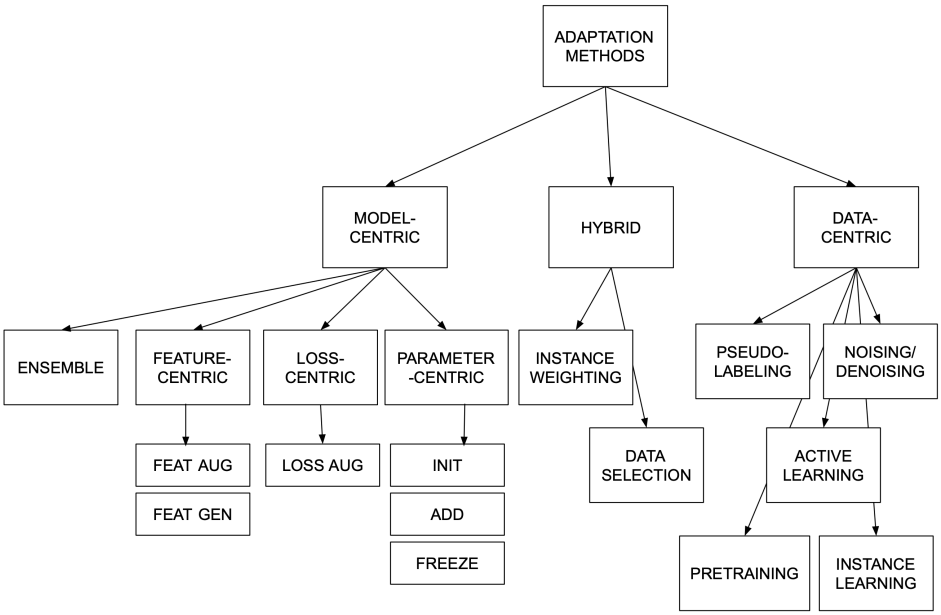


Figure 2.5: Categorization of adaptation methods proposed, extended or used in all studies.

2.2.3 Adaptation Method Categorization

For adaptation methods proposed or used in each study, we assign type labels according to the categorization presented in Figure 2.5. This categorization is an extension of the one proposed by Ramponi and Plank (2020). Since our meta-analysis is not limited to neural unsupervised domain adaptation, we need to extend their categorization with additional classes. Broadly, methods are divided into three *coarse* categories: (i) model-centric, (ii) data-centric, and (iii) hybrid approaches. Model-centric approaches perform adaptation by modifying the structure of the model, which may include editing the feature representation, loss function or parameters. Data-centric approaches perform adaptation by modifying or leveraging labeled/unlabeled data from the source and target domains to bridge the domain gap. Finally, hybrid approaches are ones that cannot be clearly classified as model-centric or data-centric. Each coarse category is divided into *fine* subcategories.

Category	Example Methods	Example Studies
Feature Augmentation (FA)	Structural correspondence learning, Frustratingly easy domain adaptation	(Blitzer et al., 2006; Daumé III, 2007)
Feature Generalization (FG)	Marginalized stacked denoising autoencoders, Deep belief networks	(Jochim and Schütze, 2014; Ji et al., 2015; Yang et al., 2015)
Loss Augmentation (LA)	Multi-task learning, Adversarial learning, Regularization-based methods	(Zhang et al., 2017; Liu et al., 2019a; Chen et al., 2020b)
Parameter Initialization (PI)	Prior estimation, Parameter matrix initialization	(Chan and Ng, 2006; Al Boni et al., 2015)
Parameter Addition (PA)	Adapter networks	(Lin and Lu, 2018)
Parameter Freezing (FR)	Embedding freezing, Layerwise freezing	(Yin et al., 2015; Tourille et al., 2017)
Ensemble (EN)	Mixture of experts, Weighted averaging	(McClosky et al., 2010; Nguyen et al., 2014)
Instance Weighting (IW)	Classifier based weighting	(Jiang and Zhai, 2007; Jeong et al., 2009)
Data Selection (DS)	Confidence-based sample selection	(Scheible and Schütze, 2013; Braud and Denis, 2014)
Pseudo-Labeling (PL)	Semi-supervised learning, Self-training	(Umansky-Pesin et al., 2010; Lison et al., 2020)
Noising/Denoising (NO)	Token dropout	(Pilán et al., 2016)
Active Learning (AL)	Sample selection via active learning	(Rai et al., 2010; Wu et al., 2017)
Pretraining (PT)	Language model pretraining, Supervised pretraining	(Conneau et al., 2017; Howard and Ruder, 2018)
Instance Learning (IL)	Nearest neighbor learning	(Gong et al., 2016)

Table 2.3: Examples of types of methods included in each category, and papers which studied these methods. These lists are non-exhaustive, but the complete method coding for all papers in our meta-analysis sample is provided in Table A.1 in appendix A.

Model-centric approaches are divided into four categories, based on which portion of the model they modify: (i) feature-centric, (ii) loss-centric, (iii) parameter-centric, and (iv) ensemble. Feature-centric approaches are further divided into two fine subcategories: (i) feature augmentation, and (ii) feature generalization. Feature augmentation includes techniques that learn an alignment between source and target feature spaces using shared features called *pivots* (Blitzer et al., 2006). Feature generalization includes methods that learn a joint representation space using autoencoders, motivated by Glorot et al. (2011); Chen et al. (2012). Loss-centric approaches contain one fine subcategory: loss augmentation. This includes techniques which augment task loss with adversarial loss (Ganin and Lempitsky, 2015; Ganin et al., 2016), multi-task loss (Liu et al., 2019a) or regularization terms. Parameter-centric approaches include three fine subcategories: (i) parameter initialization, (ii) new parameter addition, and (iii) parameter freezing. Finally ensemble, used in settings with multiple source domains, includes techniques that learn to combine predictions of multiple models trained on source and target domains.

Data-centric approaches are divided into five fine subcategories. Pseudo-labeling approaches train classifiers which are then used to produce "gold" labels for unlabeled target data. These approaches include semi-supervised learning methods such as bootstrapping, co-training, self-training, etc. (e.g., McClosky et al. (2006)). Active learning approaches use a human-in-the-loop setting to annotate a select subset of target data that the model can learn most from (Settles, 2009). Instance learning approaches include non-parametric techniques such as nearest neighbor learning which leverage neighborhood structure in joint source-target feature spaces to make predictions on target data. Noising/denoising approaches include data corruption or pre-processing which increase surface similarity between source and target examples. Finally, pretraining approaches train large-scale language models on unlabeled data to learn better source and target representations, a strategy that has gained popularity in recent years (Peters et al., 2018; Devlin et al., 2019).

Hybrid approaches consist of two fine subcategories: (i) instance weighting, and (ii) data selection. They cannot be classified as model-centric or data-centric because while they involve manipulation of the data distribution, they can also be viewed as loss-centric approaches that modify the training loss. Instance weighting includes approaches that assign weights to target examples based on their similarity to source data. Conversely, data selection approaches filter target data based on similarity to source data. Table 2.3 lists example adaptation methods for each fine category and studies from our meta-analysis subset that use these methods.

2.3 Which Long Tail Macro-Level Dimensions Do Transfer Learning Studies Target?

The first goal of our meta-analysis is to document long tail macro-level dimensions that transfer learning studies have tested their methods on. We look at distributions of tasks, domains, languages and adaptation settings studied in all papers in our sample. 10 studies are surveys, position papers or

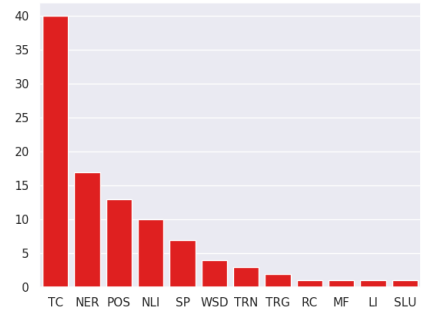


Figure 2.6: Distribution of papers according to tasks studied. The top three task categories are text classification (TC), semantic sequence labeling (NER) and syntactic sequence labeling (POS). Table 2.2 contains descriptions for the remaining task categories.

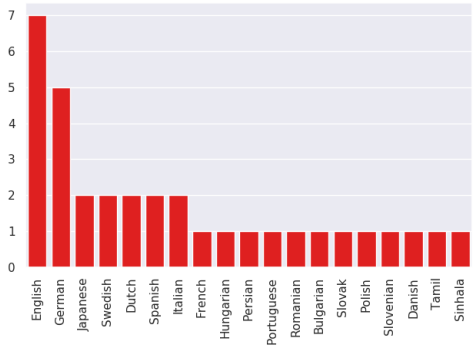


Figure 2.7: Distribution of multi-lingual studies according to languages included.

meta-experiments, and so excluded from these statistics. Studies can cover multiple tasks, domains, languages or settings so counts may be higher than 90.

Task distribution: Figure 2.6 gives a brief overview of the distribution of tasks studied across papers. Text classification tasks clearly dominate, followed by semantic and syntactic tagging. Text classification covers a variety of tasks, but sentiment analysis is the most well-studied, with research driven by the multi-domain sentiment detection (MDSD) dataset (Blitzer et al., 2007). Conversely, structured prediction is under-studied, despite covering a variety of tasks such as coreference resolution, syntactic parsing, dependency parsing, semantic parsing, etc. This indicates that tasks with complex formulations/objectives are under-explored. We speculate that there may be two reasons for this: (i) difficulty of collecting annotated data in multiple domains/languages for such tasks,[3] and (ii) shift in output structures (e.g., different named entity types in source and target domains) making adaptation harder.

Languages studied: Despite a focus on generalization, most studies in our sample rarely evaluate

[3]Note that despite these difficulties, efforts to collect data for structured prediction tasks are underway, such as the massive Universal Dependencies project which has collected consistent grammar annotations for over 100 languages: https://universaldependencies.org

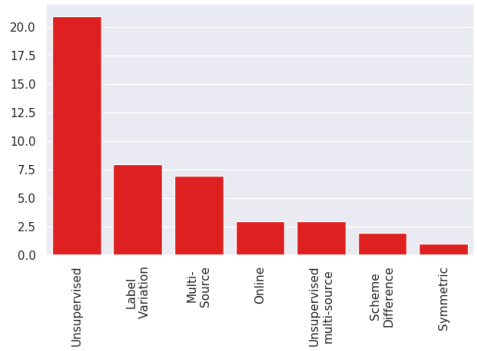


Figure 2.8: Distribution of papers according to adaptation settings studied.

Domain	#Studies
Clinical	10
Biomedical	9
Science	3
Finance	3
Literature	3
Defense & Security	1

(a) Studies on high-expertise domains.

Domain	#Studies
Twitter	12
Conversations	10
Forums	8
Emails	6

(b) Studies on non-narrative domains.

Table 2.4: Distribution of papers according to various types of long tail domains studied.

on other languages aside from English. As stated by Bender (2011), this is problematic because the ability to apply a technique to other languages does not necessarily guarantee comparable performance. Some studies do cover multi-lingual evaluation or focus on cross-linguality. Figure 2.7 shows the distribution of languages included in these studies, which is a limited subset. We note that while more non-English languages might be explored if we included "cross-lingual adaptation" work, our conclusion that non-English languages are sidelined is still valid. This is because we are trying to highlight the need for more studies that validate a monolingual adaptation technique across multiple languages. An example could be a study showing that an adaptation technique works well when transferring from English news to English conversations as well as German news to German conversations. This kind of work would not be covered under cross-lingual adaptation, and it can be concluded that this setting is under-researched. For a more comprehensive discussion of linguistic diversity in NLP research not limited to transfer learning alone, we refer interested readers to Joshi et al. (2020).

Domains studied: Many popular transfer benchmarks and datasets (Blitzer et al., 2007; Wang et al., 2019c,b) are homogeneous. They focus on expository English text, drawn from a few plentiful sources such as news articles, reviews, blogs, essays and Wikipedia. This sidelines some

key categories of domains[4] that fall into the long tail: (i) non-narrative text (e.g., social media, conversations etc.), and (ii) texts from high-expertise domains that use specialized vocabulary and knowledge (e.g., clinical text). Statistics from our meta-analysis sample support this. Tables 2.4a and 2.4b show the number of studies focusing on high-expertise domains and non-narrative domains respectively, highlighting the lack of focus on these areas.

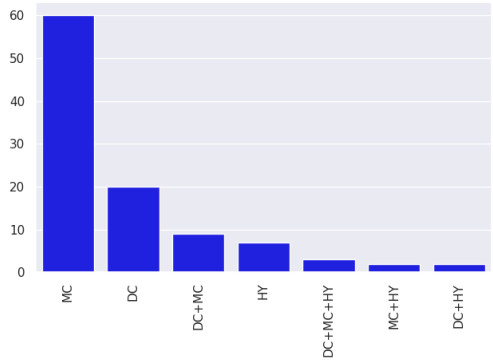
Adaptation settings studied: Most studies evaluate methods in a supervised adaptation setting, i.e. some labeled data is available from both source and target domains. This assumption may not always hold in practice. Often adaptation must be performed in harder unconventional settings such as unsupervised adaptation (no labeled data from target domain), adaptation from multiple source domains, online adaptation, etc. Figure 2.8 shows the distribution of unconventional adaptation settings across papers, indicating that these settings are understudied in literature.

Open Issues: We can see that there is much ground to cover in testing adaptation methods on the long tail. Two research directions may be key to achieving this: (i) development of and evaluation on diverse benchmarks, and (ii) incentivizing publication of research on long tail domains at NLP venues. Diverse benchmark development has gained momentum, with the creation of benchmarks such as BLUE (Peng et al., 2019) and BLURB (Gu et al., 2020) for biomedical and clinical NLP, XTREME (Hu et al., 2020) for cross-lingual NLP and GLUECoS (Khanuja et al., 2020) for code-switched NLP. However, newly proposed adaptation methods are often not evaluated on them, which is imperative to test their limitations and generalization abilities. On the other hand, application-specific or domain-specific evaluations of adaptation methods are sidelined at NLP venues and may be viewed as limited in terms of bringing broader insights. But applied research can unearth significant opportunities for advances in transfer learning, and should be viewed from a *translational* perspective (Newman-Griffis et al., 2021). For example, source-free domain adaptation in which only a trained source model is available with no access to source data (Liang et al., 2020), was conceptualized partly due to data sharing restrictions on Twitter or clinical data. Though this issue is limited to certain domains, source-free adaptation may be of broader interest since it has implications for reducing models' reliance on large amounts of data. Therefore, encouraging closer ties with applied transfer learning research can help us gain more insight into limitations of existing techniques on the long tail.

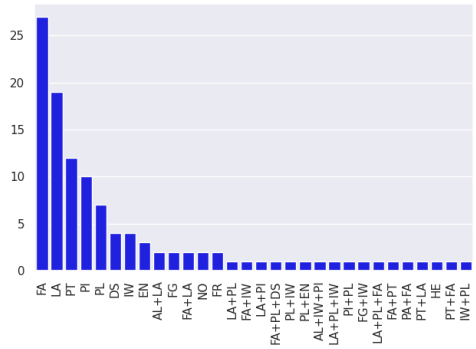
2.4 Which Properties Help Adaptation Methods Improve Performance On Long Tail Dimensions?

The second goal of our meta-analysis is to identify which categories of adaptation methods have been tested extensively, and isolate ones that have exhibited good performance on various long tail macro-level dimensions. Figures 2.9a and 2.9b provide an overview of categories of methods tested

[4]We acknowledge that "domain" is a heavily overloaded term in NLP encompassing genres, styles, registers, etc. But we use this term to remain consistent with prior literature.



(a) Distribution of transfer learning studies according to coarse method categories. DC, MC and HY refer to data-centric, model-centric, and hybrid coarse categories respectively.

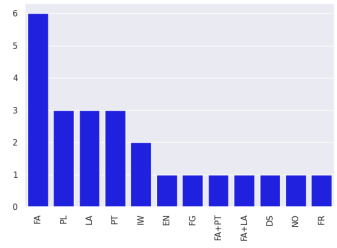


(b) Distribution of transfer learning studies according to fine method categories. The top five fine categories are feature augmentation (FA), loss augmentation (LA), pretraining (PT), parameter initialization (PI), and pseudo-labeling (PL). Table 2.3 describes the remaining categories in more detail.

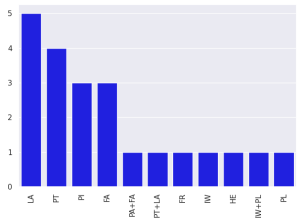
Figure 2.9: Distribution of transfer learning studies according to various types of method categories.

across all papers in our subset. We can see that studies overwhelmingly develop or use model-centric methods. Within this coarse category, feature augmentation (FA) and loss augmentation (LA) are the top two categories, followed by pretraining (PT), which is data-centric. Parameter initialization (PI) and pseudo labeling (PL) round out the top five. Feature augmentation being the most explored category is no surprise, given that a lot of pioneering early domain adaptation work in NLP (Blitzer et al., 2006, 2007; Daumé III, 2007) developed methods to learn shared feature spaces between source and target domains. Loss augmentation methods have gained prominence recently, with multi-task learning providing large improvements (Liu et al., 2015, 2019a). Pretraining methods, both unsupervised (Howard and Ruder, 2018) and supervised (Conneau et al., 2017), have also gained popularity with large transformer-based language models like ELMo (Peters et al., 2018) and BERT (Devlin et al., 2019) achieving huge gains across a variety of tasks.

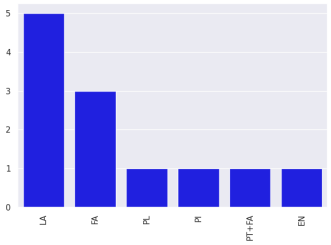
To specifically identify techniques that work on two types of long tail domains, we look at categories of methods evaluated on high-expertise domains or non-narrative domains (or both). Figures 2.10a, 2.10b and 2.10c present the distributions of fine method categories tested on high-expertise domains, non-narrative domains and both domain types respectively. While feature augmentation techniques remain the most explored category for high-expertise domains, we see a change in trend for non-narrative domains. Loss augmentation and pretraining are more commonly explored categories. The difference in dominant model categories can be partly attributed to easy availability of large-scale unlabeled data and weak signals (e.g., likes, shares etc.), particularly for social media. Such user-generated content (called "fortuitous data" by Plank (2016)) is leveraged well by pretraining or multi-task learning techniques, making them popular choices for non-narrative domains. In contrast, high-expertise domains (e.g, literature, security and defense reports, finance,



(a) Fine method categories evaluated on high-expertise domains. The top five fine categories are feature augmentation (FA), pseudo-labeling (PL), loss augmentation (LA), pretraining (PT), and instance weighting (IW). Table 2.3 describes the remaining categories in more detail.



(b) Fine method categories evaluated on non-narrative domains. The top four fine categories are loss augmentation (LA), pretraining (PT), parameter initialization (PI), and feature augmentation (FA). Table 2.3 describes the remaining categories in more detail.



(c) Fine method categories evaluated on both domain types. The top four fine categories are loss augmentation (LA), feature augmentation (FA), pseudo-labeling (PL), abd parameter initialization (PI). Table 2.3 describes the remaining categories in more detail.

Figure 2.10: Fine method categories evaluated on various types of long tail domains.

etc.) often lack fortuitous data, with methods developed for them focusing on learning shared feature spaces.

10 studies in our meta-analysis sample evaluate on both domain types. Table 2.5 describes methods explored in these studies and their performance. From these studies, we identify two interesting properties that seem to improve adaptation performance but remain relatively under-explored in the context of recent methods such as pretraining:

- **Incorporating source-target distance:** Several methods explicitly incorporate distance between source and target domain (e.g., Xing et al. (2018); Wang et al. (2018)). Aside from allowing flexible adaptation based on the specific domain pairs being considered, adding source-target distance provides two benefits. It offers an additional avenue to analyze generalizability by monitoring source-target distance during adaptation. It also allows performance to be estimated in advance using source-target distance, which can be helpful when choosing an adaptation technique for a new target domain. Kashyap et al. (2020) provide a comprehensive overview of source-target distance metrics and discuss their utility in analysis and performance prediction. Despite these benefits, very little work has tried to incorporate source-target distance into newer adaptation methods such as pretraining
- **Incorporating nuanced domain variation:** Despite NLP treating domain variation as a dichotomy (source vs target), domains vary from each other along a multitude of dimensions (e.g., topic, genre, medium or purpose of communication etc.) (Plank, 2016). Some methods acknowledge this nuanced view and treat domain variation as multi-dimensional, either in a discrete feature space (Arnold et al., 2008) or in a continuous embedding space (Yang and Eisenstein, 2015). This allows knowledge sharing across dimensions common to both source and target, improving transfer performance. This idea has also remained under-explored, though recent work such as the development of domain expert mixture (DEMix) layers (Gururangan

Study	Method	Performance
(Tsuboi et al., 2008)	Conditional random field (CRF) model trained on partially annotated sequences of OOV tokens (LA)	Positive transfer from conversations to medical manuals
(Arnold et al., 2008)	Manually constructed feature hierarchy across domains, allowing back off to more general features (FA)	Positive transfer from 5 corpora (biomedical, news, email) to email
(McClosky et al., 2010)	Mixture of domain-specific models chosen via source-target similarity features (e.g., cosine similarity) (EN)	Positive transfer to biomedical, literature and conversation domains
(Yang and Eisenstein, 2015)	Dense embeddings induced from template features and manually defined domain attribute embeddings (FA)	Positive transfer to 4/5 web domains and 10/11 literary periods
(Hangya et al., 2018)	Monolingual joint training on generic+domain text, then cross-lingual projection (PT+FA), using cycle consistency loss (Haeusser et al., 2017) (LA)	Positive transfer to medical and Twitter data using both methods
(Rodriguez et al., 2018)	Training source domain classifier and using its predictions as target classifier inputs (FA), initializing target classifier with source classifier weights (PI)	No clear winner across medical data, security and defense reports, conversations, Twitter
(Xing et al., 2018)	Multi-task learning method with source-target distance minimization as additional loss term (LA)	Positive transfer on 4/6 intra-medical settings (EHRs, forums) and 5/9 narrative to medical settings
(Wang et al., 2018)	Source-target distance minimized using two loss penalties (LA)	Positive transfer to medical and Twitter data
(Kamath et al., 2019)	Adversarial domain adaptation with additional domain-specific feature space (LA)	Positive transfer to web forums and financial text
(Lison et al., 2020)	Weakly supervised data creation by aggregating labels from rule-based or trained labeling functions (PL)	Positive transfer to financial text and Twitter

Table 2.5: Model and performance details for studies testing on high-expertise and non-narrative domains. Fine adaptation method categories used in these studies include feature augmentation (FA), loss augmentation (LA), ensembling (EN), pretraining (PT), parameter initialization (PI), and pseudo-labeling (PL).

et al., 2021) has attempted to incorporate nuanced domain variation into pretraining.

Open Issues: Interestingly many studies from our sample do not analyze failures, i.e., source-target

M \ T	TC	POS	NER	NLI	SP	WSD	TRN	TRG	RC	MF	LI	SLU
FA	25	6	13	3	4	2	1	2	1		1	1
FG	2	1	1	1	1							
LA	21	5	7	4	3		1		2		1	
PI	7	1	4	2		2	1			1		
PA			1									
FR	1	1	1									
EN	2		1	1	2							
IW	9	3	1	1		1						
DS	3		1	1	1							
PT	13		2	9	3			1			1	
PL	9	3	3	1	4				1			
NO	2	1	1									
AL	4					1						
IL	2											

Table 2.6: Evidence gap map showing indicating which method categories have not been explored sufficiently for various task categories. Please refer to Tables 2.2 and 2.3 for task and model abbreviations.

pairs on which adaptation methods do not improve performance. For some studies in Table 2.5, adaptation methods do not improve performance on all source-target pairs. But failures are not investigated, presenting the question: *do we know blind spots for current adaptation methods?* Answering this is essential to develop a complete picture of the generalization capabilities of adaptation methods. Studies that present negative transfer results (e.g., Plank et al. (2014)) are rare, but should be encouraged to develop a sound understanding of adaptation techniques. Analyses should also study ties between datasets used and methods applied, highlighting dimensions of variation between source-target domains and how adaptation methods bridge these variations (Kashyap et al., 2020; Naik et al., 2021b). Such analyses can uncover important lessons about generalizability of adaptation methods and the kinds of source-target settings they can be expected to improve performance on.

Identifying under-explored and promising methods: Annotating long tail macro-level dimensions and adaptation method categories studied by all works included in our representative sample has the additional benefit of providing a framework to identify both the most under-explored, as well as most promising methods, under various settings. Tables 2.6 and 2.7 provide evidence gap maps presenting the number of works from our sample that study the utility of various method categories on different tasks and domains respectively.[5] The first thing we note is that both maps are highly sparse, indicating that there is little to no evidence for several combinations, many of which are worth exploring. In particular, given recent state-of-the-art advances, the following settings seem ripe for exploration:

- **Parameter addition and freezing:** Though there are only four studies in our sample (providing

[5]We do not include languages since our meta-analysis does not solely focus on multilingual and cross-lingual work.

M \ D	HE	NN
FA	12	8
FG	1	
LA	9	11
PI	1	4
PA		1
FR	1	1
EN	2	1
IW	2	2
DS	1	
PT	5	6
PL	4	3
NO	1	
AL		
IL		

Table 2.7: Evidence gap map showing indicating which method categories have not been explored sufficiently for various long tail domain categories. Note that HE and NN refer to high-expertise and non-narrative domains. Please refer to Table 2.3 for model abbreviations.

positive evidence) that study parameter addition and freezing methods, we believe that given the advent of large-scale language models, these categories merit further exploration for popular task categories (TC, POS, NER, NLI, SP). Both methods attempt to improve generalization by reducing overfitting which is likely to be more prevalent with large language models, and are additionally *efficient* methods that do not require a large number of extra parameters.

- **Active Learning:** Studies included in our sample provide positive evidence for the use of active learning in an adaptation setting, but they have mainly evaluated on text classification (primarily sentiment analysis). We hypothesize that active learning during adaptation might also prove to be beneficial for task categories POS, NER, and SP, which require more complex, linguistically-informed annotation.
- **Data Selection:** Despite being similar in nature to instance weighting methods for which several studies provide positive evidence, data selection methods seem to have been under-explored. We believe that these methods might be useful for POS, NER, and SP tasks for which large-scale fortuitous data is not as easily available, and adaptation must also take into account shifts in output structure.

Despite the scarcity of both maps, there are certain method-task and method-domain combinations for which our meta-analysis sample includes a reasonable number of studies (>=10%). For these combinations, we provide a quick performance summary below:

- **Feature Augmentation:** On text classification, 12/25 studies use FA methods as baselines. Of the remaining 13 studies, 6 provide strong positive evidence, i.e., the FA method outperforms all methods across all settings/domains tested. The remaining 7 provide mixed results, i.e., there are certain domains on which this method category doesn't work best. On semantic sequence

labeling tasks like NER, 4/13 studies use FA methods as baselines, 5 show strong positive results and 4 show mixed results. Finally, on high-expertise domains, 1 study uses FA methods as baselines, 5 show strong positive results and 6 show mixed results. These observations indicate that despite their popularity, feature augmentation methods are not as strong as other method categories.

- **Loss Augmentation:** For text classification, 8/21 studies use LA methods as baselines. Of the remaining 13 studies, 11 provide strong positive evidence, while only 2 provide mixed results. On non-narrative domains, 9/11 studies provide strong positive evidence, while 2 provide mixed results. Based on their performance, loss augmentation methods seem to be extremely promising, especially for text classification and non-narrative domains.
- **Pretraining:** For text classification, 4/13 studies use pretraining as a baseline. Of the remaining 9 studies, 8 provide strong positive evidence and only one provides mixed results. Despite their relatively recent emergence, pretraining methods also seem to be extremely promising based on performance.

2.5 Which Methodological Gaps Have Greatest Negative Impact On Long Tail Performance?

The final goal of our meta-analysis is to identify methodological gaps in developing adaptation methods for long tail domains, which provide avenues for future research. Our observations highlight three areas: (i) combining adaptation methods, (ii) incorporating extra-linguistic knowledge, and (iii) application to data-scarce settings.

2.5.1 Combining Adaptation Methods

The potential of combining multiple adaptation methods has been not been systematically and extensively studied. Combining methods may be useful in two scenarios. The first one is when source and target domains differ along multiple dimensions (e.g., topic, language etc.) and different methods are known to work well for each. The second one is when methods focus on resolving issues in specific portions of the model such as feature space misalignment, task level differences etc. Combining model-centric adaptation methods, as per our categorization presented in §2.2.3, that tackle each issue separately may improve performance over individual approaches. Despite its utility, method combination has only been systematically explored by one meta-study from 2010. On the other hand, 23 studies apply a particular combination of methods to their tasks/domains, but do not analyze when these combinations do/do not work. We summarize both sources of evidence and highlight open questions.

Method combination meta-study: Chang et al. (2010) observe that most adaptation methods either tackle shift in feature space ($P(X)$) or shift in how features are linked to labels ($P(Y|X)$). They call the former category of methods "unlabeled adaptation methods" since labeled target

domain data is not needed and feature space alignment can be done using unlabeled data alone. Methods falling under the latter category require some labeled target data and are called "labeled adaptation methods". These categories do not map cleanly to specific categories in our hierarchy. Through theoretical analysis, simulated experiments and experiments with real-world data on two tasks (named entity recognition and preposition sense disambiguation), they make several interesting observations. First, they observe that combining methods generally improves performance beyond using a single method. Secondly, interaction between methods is complex, and simply combining best-performing labeled and unlabeled adaptation methods does not provide best results. Finally, when unlabeled adaptation algorithms are very strong and align source-target feature spaces well, a simple labeled adaptation algorithm such as training a model jointly on source and target data trumps more complex approaches.

Applying particular method combinations: Table 2.8 lists all studies that apply particular method combinations to their tasks/domains and fine-grained category labels from our categorization for the methods used in them. Combining methods from different coarse categories is the most popular strategy, employed by 15 out of 23 studies. Of the remaining 8 studies, 5 combine methods from the same coarse category, but different fine categories. These studies combine model-centric methods that edit different parts of the model (e.g. a feature-centric and a loss-centric method). The last 3 studies combine methods from the same fine category. Only 7 studies evaluate their method combination on at least one long tail domain.

Several studies observe performance improvements (Yu and Kübler, 2011; Mohit et al., 2012; Scheible and Schütze, 2013; Kim et al., 2017; Yang et al., 2017; Alam et al., 2018), mirroring the observation by Chang et al. (2010) that method combination helps. However, this observation is not consistent across all studies. For example, Jochim and Schütze (2014) mention that combining marginalized stacked denoising autoencoders (mSDA) (Chen et al., 2012) and frustratingly easy domain adaptation (FEDA) (Daumé III, 2007) performs worse than individual methods in preliminary experiments on citation polarity classification, which are finally omitted from the paper. Both methods are feature-centric, though mSDA is a generalization technique (FG) while FEDA is an augmentation technique (FA). Additionally, mSDA is an unlabeled adaptation technique while FEDA is a labeled adaptation technique. Owing to negative preliminary results, Jochim and Schütze (2014) do not experiment further with combination, leaving open the question of whether a different labeled adaptation technique or feature augmentation technique might have interfaced better with mSDA (or vice versa with FEDA). As another example, Wright and Augenstein (2020) show that combining adversarial domain adaptation (ADA) (Ganin and Lempitsky, 2015) with pretraining does not improve performance over pretraining alone, but combining mixture of experts (MoE) with pretraining shows improvements. Both ADA and MoE are model-centric methods, while pretraining is a data-centric method. This indicates that methods from the same coarse category may react differently in combination settings. Similarly, studies that achieve positive results do not analyze which properties of the chosen adaptation methods allow them to combine successfully, and whether this success extends to other adaptation methods with similar properties, or from the

Study	Method	LT
Different Coarse Categories		
(Jeong et al., 2009)	IW+PL	✔
(Hangya et al., 2018)	PT+FA	✔
(Cer et al., 2018)	PT+LA	✔
(Dereli and Saraclar, 2019)	FA+PT	✔
(Ji et al., 2015)	FG+IW	
(Huang et al., 2019)	PI+PL	
(Li et al., 2012)	LA+PL+IW	
(Chan and Ng, 2007)	AL+PI+IW	
(Nguyen et al., 2014)	PL+EN	
(Yu and Kübler, 2011)	PL+IW	
(Scheible and Schütze, 2013)	FA+PL+DS	
(Tan and Cheng, 2009)	FA+IW	
(Mohit et al., 2012)	LA+PL	
(Rai et al., 2010)	AL+LA	
(Wu et al., 2017)	AL+LA	
Same Coarse Categories		
(Lin and Lu, 2018)	PA+FA	✔
(Zhang et al., 2017)	FA+LA	✔
(Yan et al., 2020)	FA+LA	
(Yang et al., 2017)	LA+PL+FA	
(Gong et al., 2016)	LA+PI	
Same Fine Categories		
(Alam et al., 2018)	LA+LA	✔
(Lee et al., 2020)	PL+PL	
(Kim et al., 2017)	LA+LA	

Table 2.8: Category combinations explored by studies that combine multiple methods. LT indicates whether long tail domains were evaluated on. Fine adaptation method categories explored include feature augmentation (FA), feature generalization (FG), loss augmentation (LA), parameter initialization (PI), ensembling (EN), pseudo-labeling (PL), pretraining (PT), active learning (AL), IW (instance weighting), and data selection (DS).

same coarse/fine category.

Open questions: To fully harness the potential of adaptation method combination, we must examine the following questions further:

- Is it possible to draw general conclusions about the potential of combining methods from various coarse or fine categories?
- Which properties of adaptation methods are indicative of their ability to interface well with other methods?
- Do task and/or domain of interest influence the abilities of methods to combine successfully?

This may require more meta-studies on the method combination problem. Moreover, studies applying combinations of methods to a specific task/domain should be encouraged to delve into a deep analysis of the successes and failures they obtain.

2.5.2 Incorporating Extra-Linguistic Knowledge

Most adaptation methods leverage labeled or unlabeled text to learn generalizable representations or models. However, knowledge from sources beyond text such as ontologies, human understanding of domain/task variation, etc., can be a valuable asset to improving adaptation performance. This is especially true for high-expertise domains with expert-curated ontologies (e.g., UMLS for biomedical/clinical text (Bodenreider, 2004)). From our study sample, we observe some exploration of the following knowledge sources:

Ontological knowledge: Romanov and Shivade (2018) employ UMLS for clinical natural language inference via two techniques: (i) retrofitting word vectors as per UMLS (Faruqui et al., 2015), and (ii) using UMLS concept distance-based attention. Retrofitting hurts performance, while concept distance provides modest improvements.

Domain Variation: Arnold et al. (2008) and Yang and Eisenstein (2015) incorporate human understanding of domain variation in discrete and continuous feature spaces respectively, with some success. Table 2.5 provides method and performance details for these studies. Structural correspondence learning (Blitzer et al., 2006) also relies on manually defined pivot features common to both source and target domains, and demonstrates good performance improvements.

Task Variation: Zarrella and Marsh (2016) incorporate human understanding of knowledge required for stance detection to define an auxiliary hashtag prediction task, which improves target task performance.

Manual Adaptation: Chiticariu et al. (2010) manually customize rule-based NER models, matching scores achieved by supervised models.

Another knowledge source that is not explored by studies in our sample, but has gained popularity is providing task descriptions and some examples for sample-efficient transfer learning (Schick and Schütze, 2021). Despite initial explorations, the potential of extra-linguistic knowledge sources is largely under-explored.

Open questions: Given that availability of accurate knowledge sources differs widely across tasks/domains, it may be impractical to compare their utility in improving performance across domains. But studies experimenting with a specific source can still probe the following questions:

- Can reliance on labeled/unlabeled data be reduced while maintaining the same performance?
- Does incorporating the knowledge source improve interpretability of the adaptation method?
- Can we preemptively identify a subset of samples which may benefit from the knowledge?

2.5.3 Application to Data-Scarce Adaptation Settings

§2.3 demonstrates that most studies apply their methods in a supervised setting in which some labeled data is available from both source and target domains, in addition to unlabeled data. But availability of labeled or unlabeled data is often limited, especially for long tail domains and languages. For example, (Joshi et al., 2020) show that 2,191 languages have exceptionally limited data, making it near impossible to apply supervised adaptation. Hence, methods should also be developed for and applied to settings that reflect real-world criteria such as data availability. Data-scarce adaptation settings might be harder to perform well on, but are extremely important since they closely resemble contexts in which transfer learning is likely to be used. In particular, more evaluation should be carried out in the following data-scarce settings:

Unsupervised Adaptation: No labeled target data is available, but unlabeled data from both source and target domains is available. Sometimes, distantly supervised target data can be obtained using auxiliary resources (e.g., gazetteers, knowledge bases, etc.) and weak user-generated signals (e.g., likes, shares, etc.).

Multi-source Adaptation: Instead of a single large-scale source dataset, smaller datasets from several source domains are available (e.g., Yan et al. (2020)).

Online Adaptation: Especially pertinent for deployed models, in this setting, adaptation methods must learn to adapt to new domains on-the-fly. Often information about the target domain beyond the current sample may not be available.

Source-free Adaptation: A trained model must be adapted to a target domain without source domain data, either labeled or unlabeled. This setting is especially useful for domains that have strong data-sharing restrictions such as clinical data.

Some settings, especially unsupervised adaptation, have attracted attention in recent years. Ramponi and Plank (2020) provide a comprehensive overview of neural methods for unsupervised adaptation. In their survey on NLP for low-resource settings, Hedderich et al. (2020) cover transfer learning techniques that reduce need for supervised target data. Wang et al. (2021) list human-in-the-loop data augmentation and model updation techniques that can be used for data-scarce adaptation. However, there is room to further study performance of adaptation methods in data-scarce settings.

Open questions: Broadly, two main questions in this area still remain unanswered:

- At different levels of data scarcity (e.g., no labeled target data, no unlabeled target data, etc.), which adaptation methods perform best?
- Is it possible to identify correlations between source-target domain distance and data-reliance of adaptation methods?

This indicates the need for comprehensive meta-experiments evaluating adaptation methods in data-scarce settings.

2.6 Case Study: Evaluating Adaptation Methods on Clinical Narratives

Finally, we attempt to demonstrate how our meta-analysis framework and observations can be leveraged to systematically design case studies that can begin to provide answers to the prevailing open questions laid out in the previous section. As an example, we conduct a case study to evaluate the effectiveness of popularly used adaptation methods on high-expertise domains in data-scarce adaptation settings, a burgeoning area of interest (Ramponi and Plank, 2020). Specifically, our study focuses on the question: which method categories perform best for semantic sequence labeling tasks when transferring from news to clinical narratives, given various data-scarce adaptation settings (e.g., no labeled clinical data)? We focus on two semantic sequence labeling tasks: entity extraction and event extraction, and two data-scarce adaptation settings: no labeled target data (unsupervised), and limited labeled target data. In the limited labeled target data setting, instead of using the complete training set for the target domain, we randomly sample a subset of 1000 examples and use this as the training set (after holding out a 10% validation subset).

2.6.1 Datasets

We use the following entity extraction datasets:

- **CoNLL 2003** (Tjong Kim Sang and De Meulder, 2003): Reuters news stories annotated with four types of entities: persons (PER), organizations (ORG), locations (LOC), and miscellaneous names (MISC).
- **i2b2 2006** (Uzuner et al., 2007): Medical discharge summaries from Partners Healthcare annotated with PHI (private health information) entities of eight types: patients, doctors, locations, hospitals, dates, IDs, phone numbers, and ages.
- **i2b2 2010** (Uzuner et al., 2011): Discharge summaries from Partners Healthcare, Beth Israel Deaconess Medical Center, and the University of Pittsburgh Medical Center annotated with three entity types: medical problems, tests and treatments.
- **i2b2 2014** (Stubbs and Uzuner, 2015): Longitudinal medical records from Partners Healthcare annotated with PHI (private health information) entities of eight broad types: name, profession, location, age, date, contact, IDs, and other.

All entities are annotated in IOB format. For event extraction, we use the following datasets:

- **TimeBank** (Pustejovsky et al., 2003b): News articles from various sources annotated with events, time expressions and temporal relations between events.
- **i2b2 2012** (Sun et al., 2013): Discharge summaries from Partners Healthcare and Beth Israel Deaconess Medical Center annotated with events, time expressions and temporal relations.

i2b22006		i2b22014	
Original	**New**	**Original**	**New**
ID	MISC	ID	MISC
Doctor	PER	Name	PER
Patient	PER	Profession	MISC
Location	LOC	Location	LOC
Phone	MISC	Contact	MISC
Hospital	ORG	PHI	MISC
Date	MISC	Date	MISC
Age	MISC	Age	MISC

Table 2.9: Mappings from label sets for the i2b22006 and i2b22014 datasets to the CoNLL 2003 label set.

- **MTSamples** (Naik et al., 2021b): Medical records from the MTSamples website annotated with events. This dataset is test-only.

CoNLL 2003 and TimeBank are the source datasets for all entity and event extraction experiments respectively, while the remaining are target datasets. We focus on English narratives only. Among the NER datasets, the label sets for i2b22006 and i2b22014 can be mapped to the label set for CoNLL2003, however the label set for i2b22010 is quite distinct and cannot be mapped. Therefore, we evaluate NER in two settings: *coarse* and *fine*. In the coarse setting, the model only detects entities, but does not predict entity type, whereas in the fine setting, the model detects entities and predicts types. The coarse setting evaluation covers all target NER datasets, while the fine setting only covers datasets that can be label-mapped (i.e., i2b22006 and i2b22014). Table 2.9 presents the mapping from the label sets for i2b22006 and i2b22014 to the CoNLL 2003 label set. Note that Appendix C presents some examples of annotated instances from all these datasets.

2.6.2 Adaptation Methods

As the baseline model for NER and event extraction, we use a BERT-based sequence labeling model that computes token-level representations using a BERT encoder, followed by a linear layer that predicts entity/event labels per token. We compare the performance of adaptation methods from the top 5 fine categories most frequently applied (i.e. most popular) to high-expertise domains as per our analysis (Figure 2.10a), on top of this BERT baseline. Specific adaptation methods that we test, from each fine category, are described in more detail below:

- **FA:** Since feature augmentation (FA) methods require some labeled target data to train target-specific weights, no methods from this category are evaluated in the unsupervised setting. In the setting with limited labeled target data, we evaluate the frustratingly easy domain adaptation (FEDA) method from Daumé III (2007). This method works by creating k+1 copies of the model's feature space, comprising of one copy per domain and one

domain-general copy. During training, for each example, only the features corresponding to the example's domain and the domain-general features are populated, while all remaining features are set to 0. This structure helps the model learn which specific features are important for different domains, as well as which features are important across all domains.

- **PL:** From the pseudo-labeling category, we test the self-training method in the unsupervised setting. Self-training works by first training a sequence labeling model on the source dataset of news narratives, then using the source-trained model to generate labels for unlabeled sentences from the target domain (clinical narratives). A subset of high-confidence predictions from this set of "pseudo-labeled" clinical data are then combined with the source dataset to train a sequence labeling model. This process can be repeated iteratively until all the unlabeled data is exhausted. Unfortunately, no pseudo-labeling methods can be tested in the limited labeled target data setting because the key assumption underlying this class of methods is that we have a large corpus of unlabeled target to leverage, which is no longer true in this setting.

- **LA:** From the loss augmentation category, we test adversarial domain adaptation in the unsupervised setting (Ganin and Lempitsky, 2015). This method tries to learn domain-invariant representations by adding an adversary that tries to predict an example's domain and subtracting the loss from this adversary from the overall model loss. This setup is trained in a two-stage process, with the adversary being trained for domain prediction in the first step, and the sequence labeling model being trained to do well on sequence labeling while suppressing domain-specific information in the second step. In the limited labeled target data setting, we test multi-task training from the loss augmentation category. This method leverages the availability of labeled data from other domains (like the source domain) by training a model that consists of a shared representation learning module followed by separate task-specific layers for each domain. In our experiments, this is operationalized as training a shared BERT encoder with separate linear layers predicting entity/event labels for source and target domains. Note that losses from both domains are added together, which makes this a loss augmentation technique.

- **PT:** From the pretraining category, in the unsupervised setting, we test domain-adaptive pretraining as described by Gururangan et al. (2020). This method tries to improve target domain performance of BERT-based models by continual masked language modeling pretraining on unlabeled text from the target domain. Similar to pseudo-labeling, no pretraining methods can be tested in the limited labeled target data setting because of the key underlying assumption that we have access to a large corpus of unlabeled target data, which is not true in this setting.

- **IW:** From the instance weighting category, we test classifier-based instance weighting (e.g., Søgaard and Haulrich (2011)). In the unsupervised setting, this method trains a classifier

on the task of predicting an example's domain, then runs the trained classifier on all source domain examples and uses the target domain probabilities as weights for each example. This technique thus assigns higher weights to examples from the source domain that "look" more like the target domain according to the domain classifier, hopefully improving performance on the target datasets. In our setup, we perform interleaved training - we retrain the domain classifier after each model training pass and update the weights assigned to source dataset examples. In the limited labeled target data setting, this method first trains a target-specific classifier using the available labeled data from the target domain. This classifier is then used to relabel training data from the source domain, and then all source instances are ranked according to confidence assigned by this classifier to incorrect predictions. The weights for the bottom k examples (k is equal to the size of the target training dataset) from this ranking are set to 1, while weights for remaining examples are set to 0, discarding them from the training process.

In addition to these adaptation methods, we also evaluate the following baselines:

- **ZS:** BERT baseline model performance in a zero-shot setting, i.e., training on the source dataset (ConLL2003/TimeBank) and testing on the target dataset without any adaptation.
- **TG:** BERT baseline model performance when trained on the limited labeled target data available.
- **SC+TG:** BERT baseline model performance when trained jointly on a mixture of source domain training data and limited labeled target data.
- **SC->TG:** BERT baseline model performance when trained on source domain data, followed by training on limited labeled target data.

2.6.3 Results

Tables 2.10 and 2.11 show the results of all adaptation methods on both coarse and fine entity extraction, while Table 2.12 shows the results of all adaptation methods on event extraction.
Performance on coarse NER: From Table 2.10, we can see that in the unsupervised setting, the best-performing method categories are loss augmentation and pseudo-labeling across different datasets. Pseudo-labeling seems to work better on target datasets whose labels can be mapped to the source dataset, which can be considered *closer* transfer tasks. For i2b22010, which is the more distant transfer task, loss augmentation works best. The effectiveness of pseudo-labeling methods here is interesting because they can suffer from the pitfall of propagating errors made by the source-trained model, which may also explain their poor performance on i2b22010. Indeed, early work on applying these methods to parsing showed negative results, or very minor improvements (Charniak, 1997; Steedman et al., 2003), but these methods have shown more promise in recent years with

Setting	Model	i2b22006			i2b22010			i2b22014		
		P	R	F1	P	R	F1	P	R	F1
Unsupervised	**ZS**	18.68	21.82	20.13	35.23	10.13	15.74	21.16	**32.77**	25.71
	LA	16.11	21.17	18.30	36.60	**15.41**	**21.69**	27.50	28.56	28.02
	PL	**23.21**	22.01	**22.60**	23.26	5.03	8.28	**47.44**	23.60	**31.52**
	PT	19.50	**22.05**	20.70	**38.14**	12.75	19.11	27.25	27.35	27.30
	IW	20.98	19.53	20.23	34.31	12.12	17.91	21.00	29.22	24.43
Limited Supervision	**TG**	79.79	84.78	82.21	76.33	76.67	76.50	84.87	**84.63**	84.75
	SC+TG	80.08	74.89	77.40	78.18	70.92	74.38	79.39	64.99	71.47
	SC->TG	**86.82**	**90.39**	**88.57**	71.01	74.42	72.67	**89.12**	79.25	83.88
	FA	74.89	84.44	79.38	**81.12**	69.36	74.78	81.25	71.31	75.96
	LA	85.93	87.19	86.56	79.18	80.14	**79.66**	84.65	82.02	83.31
	IW	78.63	85.74	82.03	76.34	**80.47**	78.35	86.61	83.21	**84.88**

Table 2.10: Results of all adaptation methods on NER in the coarse setting. Unsup and Limited Sup indicate unsupervised and limited labeled target data settings respectively. Recall that the fine adaptation method categories we evaluate are feature augmentation (FA), loss augmentation (LA), pseudo-labeling (PL), pretraining (PT), and instance weighting (IW).

advances in embedding representations. In the limited labeled data setting, loss augmentation still seems to remain the best-performing method on the i2b22010 dataset. However, the best performing methods on i2b22006 and i2b22014 are the SC->TG baseline and instance weighting respectively. The high performance of instance weighting on i2b22014 is interesting because this method uses fewer examples from the source domain than other methods. Since the instance weighting method chooses the top 1000 examples from the source domain that contain the fewest proportion of high-confidence incorrect predictions according to a target-trained classifier, it carefully selects data that is deemed to be *closer* to the target domain. Its high performance despite using a small fraction of the source training data, indicates that in some cases, choosing the right subset of data is more beneficial than using all available data for adaptation.

Performance on fine NER: From Table 2.11, we can see that in the unsupervised setting, loss augmentation and pseudo-labeling method categories perform best (similar to coarse NER). Loss augmentation does better on i2b22006, while pseudo-labeling continues to be the best-performing method on i2b22014. In the limited labeled data setting, loss augmentation is still the best-performing method on i2b22006, but the SC->TG baseline is the best-performing method on i2b22014. The SC->TG baseline turns out to be particularly strong because it simultaneously leverages the availability of data from additional domains, while achieving some level of *forgetting* conflicting information from these domains by having training on the target domain be the last step in its training procedure.

Performance on event extraction: From Table 2.12, we can see that loss augmentation works best on both event extraction datasets in the unsupervised setting. Conversely, in the limited labeled data

		i2b22006			i2b22014		
Setting	**Model**	**P**	**R**	**F1**	**P**	**R**	**F1**
Unsupervised	**ZS**	12.59	14.09	13.30	23.94	**28.25**	25.92
	LA	16.08	**15.81**	**15.95**	22.77	25.70	24.15
	PL	**17.51**	11.35	13.78	**39.52**	21.36	**27.73**
	PT	10.04	12.29	11.05	17.07	22.33	19.35
	IW	14.40	14.05	14.22	21.82	25.62	23.57
Limited Supervision	**TG**	80.72	81.97	81.34	81.03	73.31	76.98
	SC+TG	78.91	75.19	77.01	78.22	63.00	69.79
	SC->TG	87.01	87.35	87.18	**87.66**	**84.30**	**85.95**
	FA	83.98	82.01	82.98	84.21	41.41	55.52
	LA	**88.45**	**88.36**	**88.40**	86.69	82.95	84.78
	IW	79.29	83.25	81.22	84.30	79.48	81.82

Table 2.11: Results of all adaptation methods on NER in the fine setting. Unsup and Limited Sup indicate unsupervised and limited labeled target data settings respectively. Recall that the fine adaptation method categories we evaluate are feature augmentation (FA), loss augmentation (LA), pseudo-labeling (PL), pretraining (PT), and instance weighting (IW).

setting, feature augmentation and instance weighting methods show comparable performance and outperform all other adaptation methods and baselines. As mentioned earlier, the high performance of instance weighting is interesting because it only uses a small subset of the source data (1000 examples) that is most similar to the target data, unlike other adaptation methods. This offers additional evidence that selecting the right subset of data to learn from can sometimes be more beneficial than using all available data (e.g., as done by the loss/feature augmentation methods). Another interesting observation is that pretraining is not the best-performing method on any dataset in the unsupervised setting. This may be a side effect of the continual pretraining process leading to some level of forgetting, which can have negative impact in an unsupervised adaptation setting. This further highlights the need to conduct such systematic studies to compare adaptation methods under data-scarce settings because the ranking of methods can change based on the availability and quality of domain-specific data. From these overall performance scores, it is clear that no single category of methods is the clear winner across all tasks and adaptation settings. Moreover, solely looking at overall performance leaves some interesting questions unanswered:

- Can specific properties of entity/event spans explain why certain adaptation methods do better for certain datasets? We are primarily interested in lexical and semantic properties of the entity and event spans.
- Is the degree of performance improvement (or degradation) achieved by various adaptation methods correlated to some distance metric between the source and target datasets?
- How does the performance of methods in the limited labeled data setting change if the number of target-specific examples is further decreased?

Setting	Model	i2b22012 P	i2b22012 R	i2b22012 F1	MTSamples P	MTSamples R	MTSamples F1
Unsupervised	**ZS**	48.78	15.28	23.27	91.40	48.01	62.95
	LA	**51.74**	**18.97**	**27.76**	88.12	**58.49**	**70.31**
	PL	44.11	11.44	18.17	**91.75**	39.33	55.06
	PT	41.46	10.36	16.57	90.15	46.32	61.19
	IW	50.46	18.08	26.62	90.56	48.39	63.08
Limited Supervision	**TG**	86.27	89.54	87.88	–	–	–
	SC+TG	86.47	89.55	87.98	–	–	–
	SC->TG	82.82	89.64	86.10	–	–	–
	FA	86.80	**90.06**	**88.40**	–	–	–
	LA	**88.31**	87.67	87.99	–	–	–
	IW	87.03	89.79	**88.39**	–	–	–

Table 2.12: Results of all adaptation methods on event extraction. Note that supervised adaptation methods cannot be tested on MTSamples, which is a test-only dataset. Unsup and Limited Sup indicate unsupervised and limited labeled target data settings respectively. Recall that the fine adaptation method categories we evaluate are feature augmentation (FA), loss augmentation (LA), pseudo-labeling (PL), pretraining (PT), and instance weighting (IW).

- Are there kinds of examples that specific adaptation methods perform well or poorly on?
- What kinds of examples does adding target domain data help adaptation methods to capture? More specifically, what kinds of examples do methods get correct in the limited labeled data setting but not in the unsupervised setting?
- What kinds of target domain examples are adaptation methods still unable to capture?

We perform additional analyses to answer these questions, as described in the following sections.

2.7 Analyses

2.7.1 Variation in Adaptation Method Performance by Span Properties

To study whether specific properties of entity/event spans affect the performance of various adaptation methods, we perform the following analyses in addition to looking at overall performance:

1. **Lexical Variation:** We look at the performance of all adaptation methods on in-vocabulary (IV) and out-of-vocabulary (OOV) spans/tokens separately. For spans, we count a span as OOV if any of the tokens in the span is an OOV token. For unsupervised settings, the vocabulary is constructed from the training and development sets for the source domain, while for limited labeled data settings, the vocabulary is constructed using training and development sets for both source and target domains.

Setting	Model	i2b22006		i2b22010		i2b22014	
		IV F1	OOV F1	IV F1	OOV F1	IV F1	OOV F1
Unsupervised	**ZS**	19.11	20.23	5.13	17.39	37.17	22.89
	LA	10.32	19.12	5.59	**24.00**	38.47	25.19
	PL	23.08	**22.55**	3.23	9.10	**38.69**	**29.01**
	PT	12.87	21.36	2.32	21.59	36.85	24.87
	IW	**24.10**	19.85	**7.67**	19.54	35.10	21.76
Limited Supervision	**TG**	85.54	81.55	78.42	74.92	79.78	87.32
	SC+TG	63.22	79.88	77.68	71.70	59.75	76.96
	SC->TG	85.07	**89.24**	74.24	71.41	**80.82**	85.37
	FA	63.70	82.73	78.25	71.94	65.37	81.16
	LA	**90.86**	85.71	**83.32**	76.78	79.81	85.17
	IW	87.06	81.06	79.92	**77.05**	79.39	**87.62**

Table 2.13: Performance of all adaptation methods trained for coarse NER on in-vocabulary and out-of-vocabulary entity spans. Unsup and Limited Sup indicate unsupervised and limited labeled target data settings respectively. Recall that the fine adaptation method categories we evaluate are feature augmentation (FA), loss augmentation (LA), pseudo-labeling (PL), pretraining (PT), and instance weighting (IW).

2. **Semantic Variation:** We look at the performance of all adaptation methods on different entity types (e.g., PER, LOC, ORG, MISC, etc.) separately. This analysis is only performed for the fine NER setting (i.e., i2b22006 and i2b22014 datasets), because we do not perform type prediction for coarse NER or event extraction.

Results from Lexical Variation Analysis

Tables 2.13 and 2.14 show the performance of all adaptation methods trained for coarse NER and event extraction on in-vocabulary (IV) and out-of-vocabulary (OOV) entities/events separately. From Tables 2.10 and 2.13, we can see that the best-performing methods for coarse NER on each dataset are also the best-performing methods on OOV entities, with the exception of i2b22010 in the limited labeled data setting. On i2b22010, loss augmentation is not the best-performing method on OOV entities, but its performance on IV entities is much higher than other methods, which might make up for the slight gap in OOV performance. We make similar observations for event extraction from Table 2.14, which shows that the best-performing method on each dataset also achieves the best performance on OOV events.

Across all datasets, loss augmentation and pseudo-labeling seem to achieve the best performance on OOV entities and events in the unsupervised setting. The high performance of loss augmentation here is unsurprising since adversarial domain adaptation explicitly tries to learn similar representations for source and target domains by suppressing their ability to be predictive of domain. However, the high performance of pseudo-labeling approaches is surprising,

Setting	Model	i2b22012		MTSamples	
		IV F1	OOV F1	IV F1	OOV F1
Unsupervised	**ZS**	19.39	26.05	74.37	52.97
	LA	**21.81**	**32.00**	**79.06**	**62.80**
	PL	15.87	19.84	70.14	41.11
	PT	15.37	17.44	72.91	50.95
	IW	20.88	30.71	73.13	54.51
Limited Supervision	**TG**	87.39	89.53	–	–
	SC+TG	87.36	90.10	–	–
	SC->TG	85.26	88.97	–	–
	FA	87.74	**90.64**	–	–
	LA	87.26	90.42	–	–
	IW	**87.98**	89.79	–	–

Table 2.14: Performance of all adaptation methods trained for event extraction on in-vocabulary and out-of-vocabulary events. Unsup and Limited Sup indicate unsupervised and limited labeled target data settings respectively. Recall that the fine adaptation method categories we evaluate are feature augmentation (FA), loss augmentation (LA), pseudo-labeling (PL), pretraining (PT), and instance weighting (IW).

and can potentially be attributed to the power of contextualized embedding representations in capturing source-target similarities. In the limited labeled data setting, instance weighting and feature augmentation seem to achieve the best performance on OOV entities and events, aside from the SC->TG baseline. Feature augmentation performing well here is also unsurprising since it explicitly partitions the feature space to learn both target-specific and domain-general information, which might offer it more flexible generalizability. The high performance of instance weighting on the other hand is interesting, because it indicates that models may sometimes achieve better generalization on the target domain when only looking at source domain examples that are most similar to the target domain. Additionally, looking at the performance of the SC->TG baseline on the datasets on which instance weighting performs well (i.e., i2b22010 and i2b22014), we can see that the lower performance of this baseline compared to the baseline trained using only target data (TG) is a strong indication that the source domain contains conflicting information, which the instance weighting method is well-suited to handle.

Results from Semantic Variation Analysis

Table 2.16 shows the performance of all adaptation methods trained for fine NER on various entity types for the i2b22006 and i2b22014 datasets. Additionally, we also report the proportions of each entity type in both datasets in Table 2.15. Note that i2b22014 does not contain any instances of the ORG entity type, based on the label mapping described in Table 2.9.

From Table 2.16, we can clearly see that the best-performing methods on each dataset according to overall score are not necessarily the best at identifying *all* entity types present in that dataset. In particular, they may not always achieve the highest performance on entity types that are less

Dataset	PER	LOC	ORG	MISC
i2b22006	25.10	12.90	2.24	59.76
i2b22014	28.01	17.33	–	54.66

Table 2.15: Proportion of various named entity types in i2b22006 and i2b22014 datasets.

frequent and do not influence overall performance as much. For example, loss augmentation is the overall best-performing method on i2b22006 in the unsupervised setting, but does not achieve highest scores on all entity types. However, it does achieve the highest performance on LOC and second highest performance on MISC (most frequent type), which boosts its performance enough to make it the best-performing method overall. Similarly, SC->TG is the best-performing method on i2b22014 in the limited labeled data setting, because it achieves second highest performance on MISC (most frequent), coupled with highest performance on PER and LOC types.

We can also observe that there isn't a clear winner for various entity types across all datasets either. For example, for the PER entity type in the unsupervised setting, instance weighting is the best-performing method on i2b22006, but loss augmentation is the best-performing method on i2b22014. In the limited labeled setting, SC->TG is the best-performing method, though its performance holds across both datasets. However, these differences may partly be an artifact of the label mapping, which does not take context into account. For example, the "hospital" category in i2b22006 is mapped to the ORG label from CoNLL 2003. However, hospital names can sometimes also be used as locations when referred to in the context of patient admissions. Since there isn't a way to automatically identify such context-specific usages short of relabeling the data manually, we do not account for context in the label mapping, which may be introducing some finer-grained confounds during type prediction.

Lastly, one interesting observation that can be made from this analysis is that adding labeled data from the target domain provides a much higher performance boost for certain entity types. Though adding labeled target data generally improves performance across all types, we can clearly see that the performance boost on the MISC type is massive (> 90 F1 points in both cases). The performance boost on the ORG type is also higher ($\sim$5x) than other types. These massive boosts can partly be attributed to the fact that some categories that fall under these types (e.g., PHI, ID, etc.) are specific to the target domain and do not appear in the source domain. Therefore, models can learn to predict these types much better if provided access to some labeled examples. This observation offers a potential solution when adapting models under data-scarce settings: focus annotation efforts on entity types that differ widely between source and target domains.

2.7.2 Correlating Domain Distance and Performance

To analyze whether the degree of performance improvement (or degradation) achieved by various adaptation methods is correlated to distance between source and target datasets, we first need

Setting	Model	i2b22006 PER	i2b22006 LOC	i2b22006 ORG	i2b22006 MISC	i2b22014 PER	i2b22014 LOC	i2b22014 MISC
Unsupervised	**ZS**	42.00	21.05	6.10	0.40	65.09	41.24	0.05
	LA	39.98	**22.71**	9.26	0.73	**69.19**	32.55	0.08
	PL	43.12	14.58	12.09	**0.79**	65.27	41.21	0.04
	PT	16.79	12.18	**16.39**	0.17	54.47	29.69	0.05
	IW	**49.02**	11.31	8.59	0.33	68.47	**41.40**	**0.10**
Limited Supervision	**TG**	72.03	0.00	44.85	95.13	69.27	45.05	89.11
	SC+TG	81.96	20.67	56.63	85.25	79.27	47.71	72.66
	SC->TG	**86.46**	**38.74**	64.73	94.18	**87.24**	**69.67**	90.02
	FA	81.00	34.88	65.81	88.69	78.23	47.35	42.12
	LA	86.13	0.00	**71.75**	95.04	85.09	64.58	**90.83**
	IW	69.31	0.00	48.71	**95.22**	83.54	50.52	89.92

Table 2.16: Performance of adaptation methods trained for fine NER on each entity type. Note that these scores are only computed for the i2b22006 and i2b22014 datasets, which can be label-mapped to the CoNLL 2003 dataset. Unsup and Limited Sup indicate unsupervised and limited labeled target data settings respectively. Recall that the fine adaptation method categories we evaluate are feature augmentation (FA), loss augmentation (LA), pseudo-labeling (PL), pretraining (PT), and instance weighting (IW).

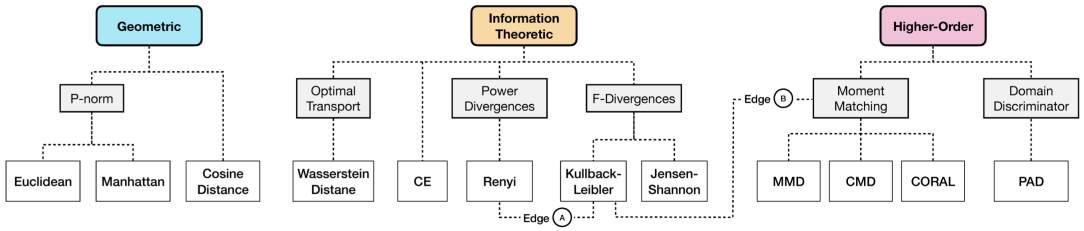


Figure 2.11: Taxonomy of various domain divergence measures developed or explored by prior work in domain adaptation, according to Kashyap et al. (2020).

to establish measures to compute source-target distance. Prior work has explored an extensive array of distance measures, motivated by the practical applicability of such measures in estimating performance drops of models on new domains (Van Asch and Daelemans, 2010) or in choosing among alternate models (Xia et al., 2020). Distance (or divergence) measures explored have ranged from linguistically-oriented measures like register variation (Biber and Conrad, 2009), to probabilistic and information-theoretic measures (Ben-David et al., 2010; Van Asch and Daelemans, 2010; Plank and van Noord, 2011) and higher-order moments of random variables (Gretton et al., 2006; Zellinger et al., 2017). Despite heavy exploration, there isn't a clear consensus on which divergence measures (or family of measures) works best for specific NLP applications or model architectures. Most recently, Kashyap et al. (2020) try to tackle this issue by surveying literature on domain divergences, developing a taxonomy of measures and performing an empirical correlation analysis to provide guidelines on choosing appropriate measures for various NLP tasks.

Figure 2.11 presents the taxonomy developed by Kashyap et al. (2020). This taxonomy divides

	CoNLL-2003			TimeBank	
	i2b22006	**i2b22010**	**i2b22014**	**i2b22012**	**MTSamples**
TVO	0.1739	0.2027	0.1629	0.1884	0.3129
KLD	2.3341	2.1348	1.9357	1.9758	2.2342
JSD	0.3896	0.3692	0.3412	0.3832	0.4403
RD	2.3125	2.1151	1.9176	1.9633	2.2239

Table 2.17: Distance between source-target domain pairs used in our experiment according to various measures. Note that TVO, KLD, JSD and RD stand for term vocabulary overlap, Kullback-Leibler divergence, Jensen-Shannon divergence and Renyi divergence respectively. As indicated in the table, for i2b22006, i2b22010 and i2b22014, distance is computed from CoNLL-2003, while for i2b22012 and MTSamples, distance is computed from TimeBank. Note that for TVO, lower values mean higher source-target distance, while higher values correspond to higher source-target distance for all other measures.

Model	**TVO**	**KLD**	**JSD**	**RD**
LA	0.8246	0.5946	0.4580	0.5948
PL	-0.3065	-0.0199	-0.2111	-0.0258
PT	-0.0492	0.2929	-0.2635	0.2763
IW	0.8670	0.5029	0.3974	0.5031
FA	0.8198	-0.3204	0.3370	-0.3104
LA	0.8676	-0.2160	-0.0964	-0.2163
IW	0.4897	-0.7460	-0.5082	-0.7471

Table 2.18: Correlation between performance improvements/drops (recorded as percentage change over baseline) and source-target domain distance for each adaptation method in both unsupervised and supervised settings. In the unsupervised setting, zero-shot scores (ZS) are used as baseline scores, while in the supervised setting, $\max$(TG, SC+TG, SC->TG) is taken as baseline score.

divergence measures into three major families as described below:

1. **Geometric:** Geometric measures calculate the distance between vector representations for source and target domains (or instances) in a metric space. While these measures are easy to calculate, they are often ineffective in very high-dimensional spaces because all distances appear the same.

2. **Information-Theoretic:** Information-theoretic measures calculate the distance between probability distributions representing the source and target domains. These probability distributions are typically distributions over word probabilities or n-gram probabilities.

3. **Higher-Order:** Higher-order measures consider matching higher-order moments of random variables or divergence in a projected space. Such measures have properties that are amenable to end-to-end learning based domain adaptation methods, which has led to them being extensively adopted in recent research.

Not all divergence measures fit neatly into this proposed taxonomy, leaving out some measures that have been used in prior work but do not have ample support (e.g., Term Vocabulary Overlap (TVO) (Dai et al., 2019)).

In addition to developing this taxonomy of divergence measures, Kashyap et al. (2020) also perform an empirical study to assess the suitability of 12 divergence measures for predicting drops in performance for three NLP tasks: POS tagging, NER and sentiment analysis. Their study only assumes a covariate source-target shift, i.e. a shift in the marginal distributions over source and target domain features, but no shifts in the label distributions. This assumption is made because measuring label distribution shifts would require access to some target labeled data, which may not always be the case. Moreover, this assumption fits well with our case study, because we are also evaluating classes of adaptation methods in an unsupervised adaptation setting. From their empirical study spanning 130 different domain adaptation scenarios, Kashyap et al. (2020) osberve that there isn't a single divergence metric that attains the best correlation scores across all tasks. However, the family of information-theoretic measures tested (namely KL-divergence, Renyi divergence and Jensen-Shannon divergence) and the TVO measure consistently provide good correlations. Although higher-order measures are not as consistent for performance drop prediction, since they are end-to-end differentiable, they are more useful for learning better representations with lower source-target distance. Based on the observations from this study, we choose TVO and information-theoretic measures for our correlation analysis between performance improvements (or drops) achieved by various adaptation methods and source-target distance. Given a source domain S and a target domain T, chosen distance measures are computed as follows:

1. **Term Vocabulary Overlap (TVO):** This measure provides an estimate of the fraction of words from the target domain vocabulary that are also present in the source domain vocabulary. It is computed as follows:

$$TVO(S,T) = \frac{|V_T \cap V_S|}{|V_T|} \tag{2.1}$$

   Despite its simplicity and inability to capture nuanced divergences between domains, Kashyap et al. (2020) found that TVO achieves strong, reliable correlations for performance drop prediction.

2. **Kullback-Leibler Divergence (KLD):** This information-theoretic measure captures the difference between the word probability distributions for the target and source domains. The source domain distribution is treated as the reference probability distribution. It is computed as follows:

$$D_{KL}(T||S) = \sum_x T(x) \log \left(\frac{T(x)}{S(x)} \right) \tag{2.2}$$

3. **Jensen-Shannon Divergence (JSD):** This measure is a symmetric extension of the KL-Divergence measure. Moreover, the square root of this measure is a metric and it can be used for non-continuous probabilities. JSD between word probability distributions of source and target domains is computed as follows:

$$D_{JS}(T||S) = \frac{1}{2}D_{KL}(T||M) + \frac{1}{2}D_{KL}(S||M) \tag{2.3}$$

$$M = \frac{1}{2}(S+T) \tag{2.4}$$

4. **Renyi Divergence (RD):** This measure is a generalization of the KL-Divergence measure, and is also called α-power divergence. Similar to KLD and JSD, RD measures the distance between word probability distributions of the target and source domains as follows:

$$D_{\alpha}(T||S) = \frac{1}{\alpha - 1}\log\left(\sum_{x}\frac{T(x)^{\alpha}}{S(x)^{\alpha-1}}\right) \tag{2.5}$$

For all information-theoretic measures (KLD, JSD and RD), we need to compute word probability distributions for both source and target domains. As described by Kashyap et al. (2020), we first filter out all stop words from source and target vocabularies and construct a joint vocabulary consisting of the top 10,000 words according to frequency across both domains. Then word probability distributions for source and target domains are computed by normalizing the occurrence counts for all words in the joint vocabulary. For RD, α is set to 0.99. Table 2.17 shows the distance scores between all source-target domain pairs considered in our case study according to these measures. It should be noted that for the TVO measure, smaller values indicate larger distance between the source and target domains, while for KLD, JSD and RD, larger values indicate larger source-target distance. From Table 2.17, we can see that while KLD, JSD and RD follow similar trends, TVO scores do not always follow the same pattern. For example, the TimeBank-MTSamples has the highest TVO score (i.e., closest source-target pair), but the KLD, JSD and RD scores for this pair are also high (i.e., distant source-target pair). This discrepancy stems from the fact that TVO only captures proportion of overlapping vocabulary, but the information-theoretic measures also capture usage frequency. Therefore, domain pairs which have a large shared vocabulary but different usage frequencies can simultaneously have high TVO scores and high KLD/JSD/RD scores.

To compute correlation between adaptation method performance and domain distance, we first compute percentage change in performance (improvement or drop) achieved by the adaptation method over a baseline. For all adaptation methods tested in the unsupervised setting, the zero-shot score (ZS) is used as baseline performance, whereas for the limited target labeled data setting, the maximum of TG, SC->TG and SC+TG scores is used as baseline performance. After computing percentage changes for each adaptation method, we calculate the Pearson correlation between these

values and the source-target distance according to each measure. Table 2.18 shows the results from this correlation analysis.

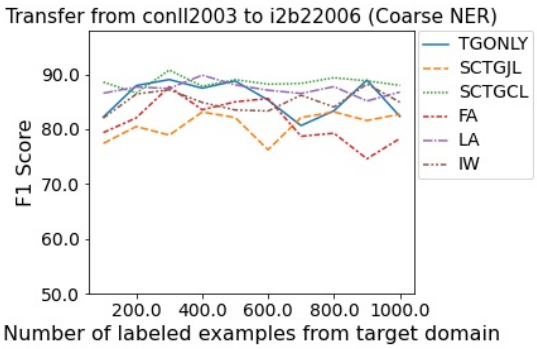
From Table 2.18, we can see that TVO scores are strongly correlated with the performance of loss augmentation and instance weighting methods in the unsupervised setting, as well as feature and loss augmentation methods in the limited target labeled data setting. This is consistent with the observation from Kashyap et al. (2020) that TVO is a strong predictor of performance drop despite its simplicity. Since higher values of TVO indicate closer source-target pairs, strong positive correlations indicate that these classes of methods provide larger performance boosts for domain pairs with a larger shared vocabulary. This is justifiable for loss augmentation and unsupervised instance weighting methods. As loss augmentation methods are designed to learn domain-general representations, especially for words occurring in both source and target domains, a larger shared vocabulary could push the model towards higher target domain performance, explaining strong positive correlations between TVO and LA. Unsupervised instance weighting uses source instances that look lexically similar to the target data to improve performance, which also allows for larger improvements given a larger shared vocabulary.

Most of the other distance measures do not achieve correlation scores as strong as TVO. However, KLD and RD do achieve strong correlation with instance weighting methods in the supervised setting. These correlations are negative, indicating that instance weighting achieves better performance gains on closer source-target pairs (lower KLD/RD). Interestingly, KLD and RD achieve moderate positive correlations with loss augmentation and instance weighting methods in the unsupervised setting. This indicates that despite varying usage across source and target domains, the presence of a large chunk of shared vocabulary is a strong indicator for larger performance boosts for these categories of methods.

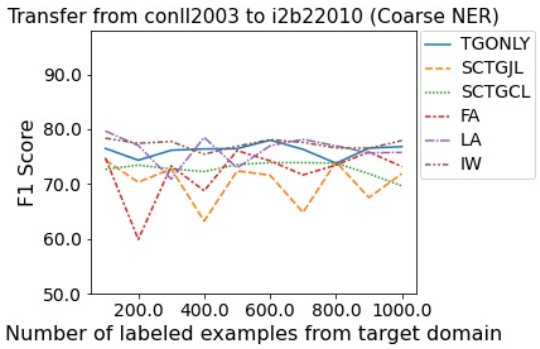
Though this analysis provides some insight into the relationship between source-target distance and adaptation method performance, it must be noted that these distance measures are not nuanced and do not capture fine-grained characteristics such as entity type distributions and syntactic constructs for words from the shared vocabulary. Moreover, though most work in NLP treats a dataset as consisting of a single domain, it is often the case that multiple domains exist within one dataset or several datasets are drawn from a single collection of texts. This has led to work questioning the *one-dataset-one-domain* assumption (Plank and van Noord, 2011). But source-target distance measures do not take this into account. Therefore, fine-grained example-level analyses such as the lexical and semantic variation analyses from the previous section must be carried out in conjunction with distance-based analyses to develop a robust understanding of the strengths and weaknesses of adaptation methods.

2.7.3 Data Reliance of Adaptation Methods

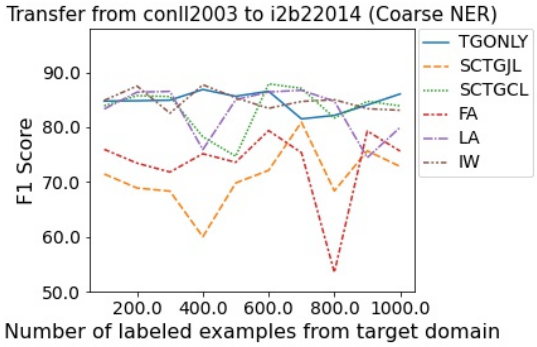
To develop a better understanding of the data dependence of various method categories in the supervised adaptation setting, we re-train all baselines and methods tested in our case study (i.e.



(a) Performance on i2b22006



(b) Performance on i2b22010



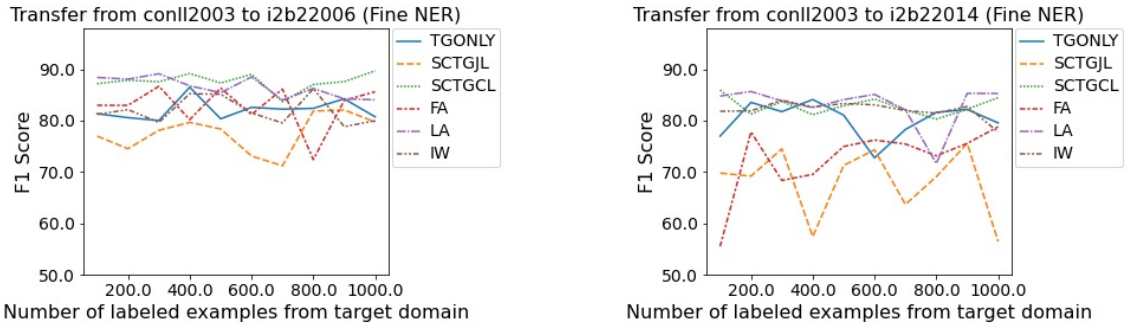
(c) Performance on i2b22014

Figure 2.12: Performance of various adaptation methods given varying number of target domain examples on coarse NER datasets. Recall that methods evaluated in a limited labeled data setting include feature augmentation (FA), loss augmentation (LA) and instance weighting (IW).

TG, SC->TG, SC+TG, FA, LA, and IW) using an increasing number of labeled examples from the target domain, and evaluate their performance at each stage. For this analysis, all methods are re-trained on subsets ranging in size from 100 to 1000 target domain examples, in increments of 100. All subsets of target domain examples are randomly sampled.

Figures 2.12a, 2.12b and 2.12c show the performance trends for all adaptation methods given target example subsets of varying sizes, on the task of coarse NER from the i2b22006, i2b22010 and i2b22014 datasets respectively. Similarly, Figures 2.13a and 2.13b show the performance trends for all adaptation methods on fine NER from i2b22006 and i2b22014 respectively. Finally, Figure 2.14 shows the performance trends for all methods on the task of event extraction from i2b22012. Note that MTSamples is a test-only dataset, and is therefore excluded from this analysis due to the lack of target domain training data. These graphs highlight several interesting points.

High starting performance: From the graphs, we can clearly see that the performance for all methods starts off with fairly high F1 scores ($\sim$2-3x the performance in an unsupervised setting), despite having access to only 100 target domain examples at that stage. Performance on event extraction from i2b22012 is particularly high. Such high starting performance can partly be



(a) Performance on i2b22006

(b) Performance on i2b22014

Figure 2.13: Performance of various adaptation methods given varying number of target domain examples on fine NER datasets. Recall that methods evaluated in a limited labeled data setting include feature augmentation (FA), loss augmentation (LA) and instance weighting (IW).

attributed to the representational power of large pretrained language models like BERT, which are able to derive utility from extremely small subsets of annotated examples. This observation also raises an important question: given a constrained budget, how should resources be divided between sourcing annotated data and applying data-scarce adaptation methods? Bai et al. (2021) explore this question for NER and relation extraction (RE) from three procedural text datasets. They observe that for small budgets, spending all funds on annotation leads to the best performance; once the budget becomes large enough, a combination of data annotation and in-domain pretraining works more optimally.

Higher performance of TG over SC+TG: Another interesting observation from the graphs is that the TG baseline consistently achieves higher performance than the SC+TG baseline on all datasets. This is interesting because the SC+TG baseline trains on source domain training data in addition to the limited number of target domain examples. Having access to more training examples could have improved the performance of this baseline over the TG baseline, but we do not see this from our analysis. This is another indication, in addition to the high performance of instance weighting methods, that adding more data is not always helpful for adaptation, especially if the data introduces conflicting signals.

High low-data performance of LA and IW: Among the domain adaptation methods, loss augmentation and instance weighting seem to provide better performance at lower data sizes over feature augmentation, in most settings. The only exception is fine NER from i2b22006. This indicates that in extremely data-scarce settings, these adaptation method categories might be the strongest contenders for initial experimentation.

Finally, noting that target domain annotated data provides a huge performance boost, even if the number of available examples is quite low, raises another interesting question: is it possible to boost performance in data-scarce settings further by choosing informative subsets of target domain examples to annotate? We explore this question in more detail in the next chapter.

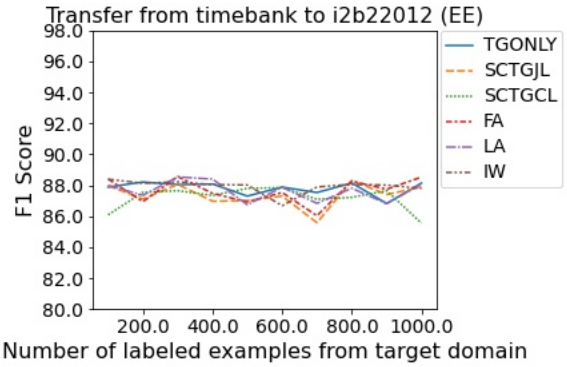


Figure 2.14: Performance of various adaptation methods given varying number of target domain examples on event extraction on the i2b22012 dataset. Recall that methods evaluated in a limited labeled data setting include feature augmentation (FA), loss augmentation (LA) and instance weighting (IW).

2.7.4 Categories of Examples Tackled by Specific Adaptation Methods

In addition to quantitative analyses studying the variation in performance of adaptation methods based on lexical and semantic properties, we also perform qualitative analyses to better identify characteristics of examples that help/hinder the performance of various adaptation methods. We believe that pairing quantitative and qualitative analyses is likely to offer deeper insight into how the performance of adaptation methods is influenced by peculiarities of the source/target data distributions, by cataloguing both broad and specific trends. Our qualitative analyses try to answer the final three questions raised by our case study.

The first question asks whether there are categories of examples that specific adaptation methods perform particularly well/poorly on, which can correspond to strengths and weaknesses of individual adaptation methods. To perform this analysis, for every adaptation method, we isolate examples that it gets incorrect for each source-target domain pair, but all other methods get correct (weakness subset), and vice versa (strength subset). Note that we analyze the coarse setting for the NER task. For each method, we examine 20 examples from each subset per domain pair, resulting in an analysis of 100 strength cases and 100 weakness cases in the unsupervised setting, and 80 strength cases and 80 weakness cases in the supervised setting (MTSamples is excluded in the supervised setting). Based on this analysis, we find some interesting differences across adaptation methods.

Observations in Unsupervised Setting:

Strengths/Weaknesses of LA: Loss augmentation methods seem to be able to deal well with vocabulary shift, and manage to accurately identify entities/events comprising of highly technical terms (e.g., “codeine”, “CXR”, “hypoglycemia”, “Apgars”, etc.). We dig deeper into this property in a subsequent case study in the next chapter. They also seem to be largely agnostic to minor orthographic differences (e.g., full capitalization of entities as in “ABDOMEN”), which other

adaptation methods find difficult to tackle. On the flip side however, these methods have a tendency to default to labeling medical terms as events/entities, especially in short sentences that do not provide much context. For example, Tamoxifen is labeled as an entity in "TAMOXIFEN 20 MG PO QD". This is especially a problem for the PHI datasets (i2b22006 and i2b22014) since they are not focused on identifying medical entities, and might partly explain why loss augmentation methods lose out to pseudo labeling on coarse NER from these datasets. Another weakness on the event extraction task seems to be the tendency to annotate some entities as events because the head noun has a second (potentially more common in source domain) word sense that can be used as an event verb (e.g., annotating "meeting" in the phrase "cervix meeting").

Strengths/Weaknesses of PL: Strength cases for pseudo-labeling methods primarily consisted of sequences with no entities/events, making it difficult to identify characteristics of entity spans that this method performed well on. The only exception was the MTSamples dataset, and most strength cases from this dataset were simple verbs indicating past/present occurrences (e.g., noted, examined). Looking at weakness cases however revealed that these methods struggle with medical vocab (e.g., "morphine", "prolapse", "extubated", etc.), which is reflected in their poor performance on i2b22010. This behavior is in strong contrast to the loss augmentation class of methods. It is also interesting to note that despite struggling with medical vocabulary, this class of methods can achieve high performance if the *task* is not centered around medical entities/events as in i2b22006 and i2b22014 (PHI identification).

Strengths/Weaknesses of PT: Similar to pseudo-labeling methods, strength cases primarily consisted of sequences with no entities/events for i2b22006, i2b22014 and i2b22012. However, on i2b22010 and MTSamples, we observed that some instances of highly technical phrases were correctly identified as entities/events (e.g., "hypertension", "capsulectomy", "friability", etc.), indicating that this class of methods captures medical vocabulary better than pseudo-labeling. However, these methods are not as agnostic to orthographic differences like capitalization, and also share the tendency of loss augmentation methods to predict medical terms as entities in the absence of much surrounding context. Interestingly, on event extraction from MTSamples, many stative events are missed, and instructional verbs (e.g., "see" in "please see") are mistakenly marked as events.

Strengths/Weaknesses of IW: Again, strength cases primarily consisted of sequences with no entities/events, with the exception of MTSamples. On MTSamples, instance weighting methods were able to identify a few medical terms, but majority cases were primarily events representing past/present occurrences and short-term stative events. Looking at weaknesses showed inability to handle orthographic differences like capitalization was a definite issue with this class of methods, however no other phenomena stood out clearly.

Observations in Supervised Setting:

In comparison to the unsupervised setting, analyzing strength and weakness cases for adaptation methods in the supervised setting yielded fewer method-specific observations. Strength cases

for most methods included some MD name, date and time entities, as well as highly complex medical terminology, but as we demonstrate in the subsequent analysis, improved performance on these phenomena can be attributed more to the availability of labeled target data. However one aspect on which both loss augmentation (LA) and instance weighting (IW) methods seemed to do slightly better than feature augmentation (FA) was their ability to handle longer spans and longer entity/event lists (e.g., "an embolus in the right profunda / femoral artery", patient denied "nausea" , "vomitting" , "abdominal pain" , "dysuria" , "dizziness"). Both IW and LA methods were also able to accurately identify boundaries when tackling consecutive entities (e.g., place followed by time). Lastly, on event extraction, LA methods suffered from the problem of ignoring adjective descriptors (e.g., only labeling "hypotensive" instead of "mildly hypotensive", or "a Doppler signal" instead of "a strong Doppler signal").

This analysis provides examples of some linguistic phenomena, in addition to vocabulary handling, that specific methods are able to handle better than others. Many of these phenomena (e.g., capitalization differences, long lists, etc.) are relatively rare and therefore might not influence overall task performance or method ranking. However, identifying such connections between phenomena and methods can still be helpful, especially when trying to adapt to a new domain where these rare phenomena are more common (e.g., social media text).

2.7.5 Categories of Examples That Benefit from Adding Target Labeled Data

Our second qualitative analysis focuses on identifying categories of examples on which we observe improved performance after adding limited labeled data from the target domain. To isolate these examples from each dataset, we collect the set of examples that *all* methods and baselines get wrong in the unsupervised adaptation setting, but right in the supervised adaptation setting. We hope that choosing this set of examples will minimize the possibility of including examples that specific methods do not fare well on. From this set of examples for each dataset, we randomly sample 50 examples for our final qualitative analysis. Note that since the MTSamples dataset does not have in-domain training data, we cannot test supervised adaptation methods on it, and hence it is excluded from this qualitative analysis. For the NER datasets, we perform this qualitative analysis for the coarse setting.

Table 2.19 gives a brief overview of the error categories, along with example instances, that we observe from our qualitative analysis of the three NER datasets (i2b22006, i2b22010 and i2b22014). From the table, we can see that several error categories arise due to entity writing formats being slightly different in medical narrative text (e.g., DAT, NUM, NAM, LOC). In particular, the DAT category is interesting because temporal entities annotated in i2b22006 and i2b22014 usually only include dates, days and months and do not annotate years as entities. This is because years are not considered protected attributes under HIPAA, and therefore not scrubbed during PHI removal. This is different from typical NER, in which years would also be included as part of temporal

Category	Description	Examples
DAT	Date entities (or boundaries) incorrect	01/12 /1992 12:00:00 AM
NUM	Numerals (or boundaries) incorrect	081039790 EH
NAM	Name entities (or boundaries) incorrect	ANA V. A, M.D.
LOC	Locations not annotated in gold data	She was received in the Respiratory Intensive Care Unit immediately.
MVE	Entities consisting of medical terms	Lipase and amylase remained normal.
MPE	Phrases that undergo meaning change when used in medical contexts	There was no obstruction.
ABR	Abbreviated entities	hh / bmot
NTE	Not entities per dataset guidelines (i.e. entities from types not included in data)	She had a DVT in 11 /96 on Coumadin preoperatively.

Table 2.19: Error categories observed from an analysis of examples from NER datasets, which are tagged correctly on adding target domain labeled data. Note that yellow highlights indicate gold entities, while pink highlights indicate entities identified by unsupervised adaptation methods that are not present in gold data.

entities. Such minor inconsistencies in annotation guidelines can lead to performance drops on an entire category of entities, and qualitative analyses can help identify such scenarios. A potential solution for such annotation inconsistencies could be to identify and annotate instances from that specific category to update the model. Aside from format-dependent categories, several error categories also arise from incomplete understanding of medical vocabulary (MVE, MPE, ABR), particularly abbreviations and terms that undergo semantic drift (i.e., change meanings when used in medical contexts). The last error category (NTE) includes spans that are identified as entities by our models, but are not annotated in the gold data because the dataset does not include entities from that category. For example, Table 2.19 shows an example from the i2b22006 dataset, in which the entity "Coumadin" is identified. It is a medical entity (drug), however since this dataset focuses only on entities that would count as PHI, "Coumadin" is not included. Learning such task-specific distinctions is another scenario in which having some labeled data can help update the model appropriately. Table 2.19 shows how many examples fall into each category for all three NER datasets. As expected, format-related errors are much higher on i2b22006 and i2b22014, both of which focus on PHI removal, while i2b22010, which focuses on medical entities, has a higher proportion of errors arising from medical terminology issues. Lastly, we note that entity boundary conditions are not satisfied in a small percentage of cases (for example, having an I-ENT tag without a B-ENT tag before it). Such issues arise because our model architecture does not enforce this boundary condition since it does not have sequential dependencies among predicted tags, and can be resolved using a CRF layer.

Table 2.21 similarly gives a brief overview of the error categories, along with example instances and number of examples in each category, observed from a qualitative analysis of the i2b22012 event extraction dataset. Unlike the NER datasets, none of the error categories here arise due

Dataset	DAT	NUM	NAM	LOC	NTE	MVE	MPE	ABR
i2b22006	16	16	3	4	9	–	–	4
i2b22010	–	–	8	1	–	33	11	1
i2b22014	26	2	6	–	25	–	–	1

Table 2.20: Proportion of errors from each category observed from an error analysis of 50 randomly sampled cases from each NER dataset, which are tagged correctly on adding target domain labeled data.

Category	Description	Examples	Num
AVE	Associated verb (or adjective) annotated by model in place of noun	Pt was continued on prophylactic heparin.	21
MVE	Verb events not present in gold data	It was felt that, because of evidence	6
MNE	Non-event nouns according to source guidelines	This was placed on postoperative day no. 9 without any difficulty.	2
ENT	Entities according to source domain guidelines	The patient was admitted to the Intermediate Care Unit	7
MTE	Events containing medical terms	He was admitted for anticoagulation and hemodynamic monitoring.	19
UNK	Unknown cause for error	DISCHARGE DATE :	6

Table 2.21: Error categories observed from an analysis of examples from the i2b22012 event extraction dataset, which are tagged correctly on adding target domain labeled data. Note that yellow highlights indicate gold events, while pink highlights indicate events identified by unsupervised adaptation methods that are not annotated in gold data.

to different writing formats. Most of the error categories (AVE, MVE, MNE, ENT) occur due to discrepancies between what is considered an event in the source and target data annotation guidelines. For example, consider the sentence "patient was continued on heparin". According to the source dataset guidelines, "continued" would be considered the event since that is the word referring to the action/accomplishment/state being discussed. However, the i2b22012 dataset, in an attempt to make annotated events medically informative, marks the associated treatment noun "heparin" as the event. There are several such categories of spans that would ordinarily be considered entities but are marked as events according to the i2b22012 guidelines (e.g., "Intermediate Care Unit" in row 4). Again, bridging this gap between guidelines is difficult to do in a completely unsupervised fashion, since the model does not know how label distributions are shifting. Aside from guideline-related categories, we again see examples that require better understanding of clinical vocabulary (MTE) forming a large proportion of error cases. From this analysis, we can see that adding target domain labeled data has great utility for spans with major variation in format, terminology and labeling, providing a small set of annotated examples.

Category	Description	Examples
BIE	Boundary inconsistency, especially for long and consecutive entities, and lists	ICECA NIGHT , M.D. 368 PBE-2ND FLOOR OFM 070 PQK 960
TIM	Temporal phrases missed by models	New Years Eve
MVE	Medical terms that aren't entities	ANGIOSARCOMA
ABR	Abbreviated entities	cut-off/combined 5/3 /99 jq
NTE	Not entities per dataset guidelines (i.e. entities from types not included in data)	She had a 1:1 sitter at all times.
ABE	Ambiguous phrases as entities	Cardiothoracic Surgery was on standby
UNK	Unknown cause for error	PT not at PROMPTCARE.

Table 2.22: Error categories observed from an analysis of examples from NER datasets, which are tagged incorrectly by all supervised adaptation methods (and baselines). Note that yellow highlights indicate gold entities, while pink highlights indicate entities identified by unsupervised adaptation methods that are not present in gold data.

Dataset	BIE	TIM	MVE	ABR	NTE	ABE	UNK
i2b22006	32	4	4	4	–	–	8
i2b22010	30	–	–	1	6	3	12
i2b22014	27	–	–	2	–	–	22

Table 2.23: Proportion of errors from each category observed from an error analysis of 50 randomly sampled cases from each NER dataset, which are tagged incorrectly by all supervised adaptation methods and baselines.

2.7.6 Categories of Examples Still Left Out: The Long Tail to the Long Tail

Our final qualitative analysis aims to identify the long tail to the long tail, i.e. what categories of examples do all methods still get wrong even after having access to some labeled data? The goal of this analysis is to identify key phenomena that adaptation methods are still unable to capture, at a more micro-level even if this analysis cannot span all examples. For this analysis, we collect all examples that *all* supervised methods and baselines get wrong, and then randomly sample 50 examples for each dataset. As in the previous analysis, MTSamples is excluded due to lack of in-domain training data for supervised adaptation.

Table 2.22 summarizes the error categories arising from our qualitative analysis of the three NER datasets (i2b22006, i2b22010 and i2b22014), and provides examples for each. From the table, we can see that some of these categories (NTE, ABR) were also observed in the previous qualitative analysis, indicating that despite adding pertinent training data, these error categories are not perfectly resolved. However, we also see several new categories that highlight phenomena that are still out of reach for current adaptation methods. One of these categories is boundary

Category	Description	Examples	Num
BIE	Boundary inconsistency for long/consecutive events, and lists	DM - glipizide , ISS	35
MSE	Events missed (cause unclear)	She was also changed to intravenous Solu-Medrol.	5
EXE	Extra events found (cause unclear)	WBCs high but normalized.	13
ABR	Abbreviated events	CMED service was consulted.	3

Table 2.24: Error categories observed from an analysis of examples from the i2b22012 event extraction dataset, which are tagged incorrectly by all supervised adaptation methods (and baselines). Note that yellow highlights indicate gold entities, while pink highlights indicate entities identified by unsupervised adaptation methods that are not present in gold data.

inconsistency errors (BIE), which especially arises when sentences contain long entities (e.g., addresses), entities in consecutive positions or lists of entities (e.g., medications). This is an interesting category of errors because it requires knowledge about general formats like addresses, as well as more domain-specific formats like a listing of observations from a physical examination, and is also highly contextual. This makes it a category that is likely to vary highly across domains and instances. Another category of errors is medical terms that appear in isolation (e.g. row 3) and in the absence of surrounding context, models default to labeling them as entities. However, they are not annotated as entities in the gold data. A third category of errors is temporal phrases that are missed by models because they make indirect references to dates (e.g., new years eve). Finally, the last non-unknown category of errors consists of ambiguous phrases (ABE) for which type disambiguation is difficult even in context, making it hard to decide whether the phrase is a entity under the current dataset/task scope. For examples, in the sentence "Cardiothoracic Surgery was on standby", the phrase "cardiothoracic surgery" likely does not refer to the procedure, but the department, and is therefore not an entity in i2b22010 (which focuses on medical entities). Table 2.23 shows how many examples fall into each category for all NER datasets. From this table, it is clear that boundary inconsistency is the most prevalent error type in this setting, followed by examples for which the cause of error cannot be deduced.

Table 2.24 presents the error categories arising from our qualitative analysis of the i2b22012 event extraction datasets, along with example instances and number of examples falling into each category. From the table, we can see that similar to NER, boundary inconsistency is a huge problem for event extraction as well, with nearly 70% of the errors falling into this category. We see that abbreviations continue to pose a problem even in this setting. For all remaining errors, though they can broadly be divided into two categories based on whether the model is missing events or annotating extra events, it is difficult to identify a pattern or phenomenon that causes the model to fail.

From this analysis, it is clear that while adding some labeled target domain data helps bridge the performance gap significantly, there still remain categories of examples that models fail on. It is

interesting to note that some categories require highly instance-specific, contextualized reasoning, which may raise the question: do models trained in a maximum likelihood estimation (MLE) paradigm have the capacity to do well on such instances?

2.8 Conclusion

This chapter presented a two-level conceptualization of the long tail, and a qualitative meta-analysis of 100 representative papers on domain adaptation and transfer learning in NLU, with the aim of understanding the performance of adaptation methods on the long tail. Through this analysis, we assessed current trends and highlighted methodological gaps that present major avenues for future research in transfer learning for the long tail. We observe that current research has a tendency to sideline certain types of tasks, languages, domains, and adaptation settings, indicating that long tail coverage is far from comprehensive. We also identify two properties that help long tail performance, but have not received much attention in recent adaptation research: (i) incorporating source-target domain distance, and (ii) incorporating a nuanced view of domain variation. Additionally, we identify three major gaps that must be addressed to improve long tail performance: (i) combining adaptation methods, (ii) incorporating extra-linguistic knowledge and (iii) application to data-scarce adaptation settings. Finally, we demonstrate the utility of the framework and observations resulting from our meta-analysis in guiding the design of systematic meta-experiments to address prevailing open questions by conducting a systematic evaluation of popular adaptation methods for a high-expertise domain (clinical text) in a data-scarce setting. This case study revealed interesting insights about the adaptation methods evaluated, highlighted key questions that need to be studied further, and showed that significant progress can be made towards developing a better understanding of adaptation for the long tail by conducting such experiments.

CHAPTER 3

Improving Macro-Level Adaptation: A Case Study on Event Extraction

Chapter 2 presented a qualitative meta-analysis of representative work on transfer learning for NLU, with an eye towards categorizing and understanding the performance of adaptation methods on macro long tail domains, through a case study on two semantic sequence labeling tasks (NER and event extraction) for clinical narratives. In this chapter, we delve deeper into the problem of data-scarce adaptation between macro-level long tail dimensions for the task of event extraction, for which building a high-performing generalizable system has remained an elusive goal. We propose two new adaptation methods:

- Likelihood-based instance weighting (LIW) (Naik et al., 2021b)
- Active learning with domain-aware query sampling (DAQ)

LIW is an unsupervised adaptation method, while DAQ, being an active learning method, requires small amounts of labeled data from the target domain. As in the previous chapter, we evaluate the performance of both methods on clinical narrative datasets. In addition to clinical narratives, we bring two additional domains under our purview: (i) doctor-patient conversation transcripts (both high expertise and non-narrative) (Naik et al., 2021b), and (ii) literary texts (a high expertise domain) (Sims et al., 2019). From our experiments, we see that LIW improves performance over a zero-shot baseline, and while it is not the best-performing method on the domains tested, it performs best on certain categories of examples (e.g., event types not present in source data). Similarly we see that DAQ improves performance over active learning baselines. Though these gains are modest for event extraction, additional experiments on NER demonstrate that DAQ can be quite powerful in certain settings. Interestingly, most active learning variants do not outperform a random sampling baseline indicating limited utility of incorporating active learning in an adaptation setting. These

case studies expand the set of domains and settings studied so far, identify strengths and weaknesses of our proposed techniques, and further our understanding of the performance of various classes of adaptation methods on macro long tail domains.

3.1 Introduction

Events are an important phenomenon in the field of computational semantics. They offer an intuitive mechanism for constructing structured representations of text, which can be used for downstream tasks such as question answering and summarization. Events also embody a crucial function of language: the ability to report happenings. Narratives from many diverse domains (e.g., news articles, literary texts, clinical notes) use events as basic building blocks. These characteristics make event extraction a key sub-task of interest for text understanding pipelines in multiple domains, including high expertise domains such as clinical notes and scientific articles. Despite the importance of this task, building high-performing and generalizable systems for event extraction has remained an elusive goal.

One of the major hurdles is that the notion of what counts as an *important event* is usually task-specific or domain-specific (sometimes both). For example, to build a system that can track a patient's disease progression from clinical notes, event extractors only need to focus on extracting medical events relevant to that illness. This task/domain specificity has encouraged prior work to focus on specific event types (Grishman and Sundheim, 1996; Doddington et al., 2004; Kim et al., 2008) or domains (Pustejovsky et al., 2003b; Sims et al., 2019), leading to a heavy emphasis on building datasets/tools for populous domains such as news articles. Owing to this narrow focus, and the abundance of annotated datasets built from news data, supervised event extraction models often fail to adapt to new domains or event types (Keith et al., 2017), especially domains that fall into the macro long tail. Conversely, unsupervised event extractors that use syntactic rule-based modules (Saurí et al., 2005; Chambers et al., 2014) have a tendency to over-generate by labeling most verbs and nouns as events, diminishing their applicability to new domains. This current state of affairs makes the event extraction task an interesting testbed to study macro-level adaptation from a theoretical perspective. Additionally, developing methods for better macro-level adaptation of event extractors has immediate and widespread practical utility.

In this chapter, we tackle the task of building event extractors that are more *adaptable* at the macro-level in a data-scarce setting, i.e. there is no or very little annotated training data from the domain of interest. To achieve this, we propose two new macro-level adaptation methods:

- Likelihood-based instance weighting (LIW) (Naik et al., 2021b)
- Active learning with domain-aware query sampling (DAQ)

The first method, likelihood-based instance weighting (LIW), is an unsupervised adaptation technique, which uses no labeled target data, that uses language model likelihood scores to reweight source domain instances based on their similarity to target domain instances. LIW falls under the hybrid coarse category and the fine category of instance weighting in our adaptation method

taxonomy. This reweighting results in better alignment between the marginal distributions of the source and target domains that we are transferring between. From a machine learning perspective, this method aims to tackle covariate shift, i.e. shift between marginal distributions of source and target domains, which is a key contributor to overall distributional shift (Ben-David et al., 2010). From a linguistic perspective, we hope that using language model scores to reweight source instances pushes this method to leverage word contexts, in addition to relying on the vocabulary shared between source and target domains, like most methods studied in the previous chapter.

The second method, active learning with domain-aware query sampling (DAQ), is an active learning technique that incorporates distance from source domain instances into the sampling criterion when choosing instances from the target domain to annotate. This method can optionally be used in conjunction with unsupervised adaptation to incorporate limited amounts of labeled data in a more sample-efficient manner. DAQ falls under the coarse category of data-centric methods and the fine category of active learning according to our adaptation method taxonomy. The goal of incorporating source domain distance as an additional term is to ensure that active learning avoids selecting target instances that are similar enough to the source domain that we could expect a model trained on the source data to do reasonably well on them already. We explore two formulations of source domain distance: (i) cosine similarity of an instance with source domain instances in an embedding space (DAQ-CS), and (ii) probability odds ratio of an instance belonging to the target domain according to a classifier trained to separate source and target instances (DAQ-CC). From a machine learning perspective, this attempts to further optimize the *informativeness* of the target instances chosen for labeling at each iteration (Rai et al., 2010). From a linguistic perspective, we again hope that using language models to develop embedding spaces (as in Aharoni and Goldberg (2020)) and domain separation classifiers encourages reliance on word contexts, in addition to shared vocabulary. Additionally, we use this as an opportunity to study sample-efficiency of active learning methods in a limited labeled data setting, which was not explored much in the previous chapter.

This chapter presents experiments evaluating the effectiveness of both LIW and DAQ, and analyses comparing their performance with other strong adaptation methods on the task of event extraction. Of various task formulations used widely, we adopt the formulation of event extraction as the task of labeling **triggers**, i.e., words which instantiate an event. For example, in the sentence "She was diagnosed with cancer," *diagnosed* and *cancer* are triggers, referring to "diagnosis" and "illness" events respectively. Throughout this chapter, we model event trigger labeling as token-level classification. Since we are interested in bringing more macro long tail domains under our purview, especially one from the non-narrative category, we create new event extraction test sets for two medical domains: (i) clinical records, and (ii) doctor-patient conversation transcripts. To do so, we develop comprehensive event annotation guidelines, based on TimeML (Pustejovsky et al., 2003a) and Thyme-TimeML (Styler IV et al., 2014). Using these guidelines, we annotate 45 documents from each domain to create new test sets.

To evaluate the effectiveness of LIW, we perform a case study according to the following

experimental setup:

- **Task:** Event extraction (semantic sequence labeling)
- **Source Domain:** News articles (TimeBank) (Pustejovsky et al., 2003b)
- **Target Domain(s):** Clinical notes (MTSamples), Doctor-patient conversation transcripts (Abridge) (Naik et al., 2021b)
- **Task Model:** BERT-BiLSTM event extractor (Sims et al., 2019)
- **Adaptation Method:** Likelihood-based instance weighting (LIW; hybrid method)
- **Adaptation Baseline(s):** Adversarial domain adaptation (ADA; model-centric method) (Ganin and Lempitsky, 2015), Domain adaptive finetuning (DAFT; data-centric method) (Han and Eisenstein, 2019)
- **Adaptation Setting:** Unsupervised

We also note that ADA is task-guided since it jointly performs alignment and task training. On the other hand, DAFT and LIW are task-agnostic, performing alignment and task training sequentially. In addition to overall performance, we also analyze the behavior of these methods under various types of covariate shifts (e.g., lexical shift, event type shift) to gain insight into differences between them. Our results show that DAFT and LIW (our method) improve over the BERT-BiLSTM baseline on both domains, whereas ADA only improves on clinical notes. Across both domains, there is no clear winner, with ADA and DAFT performing best on notes and conversations respectively. Analyzing covariate shift at different levels (e.g., lexical shift, event type shift), we uncover interesting patterns such as the ability of models to leverage sub-word morphology to generalize to some technical terms in clinical notes, and LIW's performance improvement on long-term state events which are truly zero-shot since they never appear in the source data (e.g., chronic illnesses). Interestingly, our best models achieve F1 scores of 70.0 and 72.9 on notes and conversations respectively with *no* training data.

To evaluate the effectiveness of DAQ, we perform a case study according to the following experimental setup:

- **Task:** Event extraction (semantic sequence labeling)
- **Source Domain:** News articles (TimeBank) (Pustejovsky et al., 2003b)
- **Target Domain(s):** Clinical notes (i2b22012) (Sun et al., 2013), Literary texts (LitBank) (Sims et al., 2019)
- **Task Model:** BERT-MLP event extractor
- **Adaptation Method:** Active learning with domain-aware query sampling (DAQ; active learning method)
- **Adaptation Baseline(s):** Uncertainty sampling with representativeness (UNS; active learning method) (Liao and Grishman, 2011), Query-by-committee (QBC; active learning method) (Settles and Craven, 2008)
- **Adaptation Setting:** Limited supervision

Our experiments show that adding the domain-awareness criterion during sampling helps improve the performance of active learning baselines, though these gains are quite modest ($\sim$ 1 F1 point). To further map out the utility of this criterion by studying whether it provides larger

improvements on other tasks, we conduct additional experiments on named entity recognition from clinical narratives, using the same set of datasets as the previous chapter (i2b2 2006, i2b2 2010, and i2b2 2014). On these tasks, we see much larger gains from adding domain-awareness ($\sim 4-18$ F1 points). Of the two formulations that we experiment with, the similarity-based formulation (DAQ-CS) tends to achieve better performance across most settings. We also perform a correlation analysis with label-aware variants of source-target divergence measures and observe that domain-awareness helps bridge larger label drifts (i.e., changes in word type-label associations). However, ultimately none of the active learning variants are able to outperform a simple random sampling strategy. This helps us establish a potential new failure case for active learning methods in a domain adaptation setting.

Ultimately, these case studies expand the set of dimensions studied for macro-level adaptation by including new domains, methods, and adaptation settings, and help us obtain additional evidence (both positive and negative) for our observations from the previous chapter.

3.2 Background

3.2.1 Event Extraction

Most prior event extraction work has focused on news articles, resulting in the development of several datasets (Onyshkevych et al., 1993; Grishman and Sundheim, 1996; Pustejovsky et al., 2003b; Doddington et al., 2004; Lee et al., 2012; Cybulska and Vossen, 2014; Mitamura et al., 2016). Recently, event extraction has also been explored in other domains such as biology (Wattarujeekrit et al., 2004; Kim et al., 2008, 2009; Berant et al., 2014), Wikipedia articles (Araki and Mitamura, 2018), social media data (Ritter et al., 2012; Li et al., 2014; Jain et al., 2016) and literary novels (Sims et al., 2019). Aside from data domain, event extraction paradigms (both datasets and tools) differ along three major axes: (i) event extraction granularity, (ii) event representation, and (iii) event categorization (ontology). We briefly describe these axes to contextualize our choice of event paradigm.

Event extraction granularity divides extraction paradigms into two types: (i) document-level paradigms which assume that a piece of text refers to a single event (Grishman and Sundheim, 1996), and (ii) sentence-level paradigms which assume that a single sentence describes one or more events. Event representation divides extraction paradigms into two types: (i) span-based paradigms which represent events by marking text spans that refer to events, called **triggers** or **nuggets** (Pustejovsky et al., 2003a; Mitamura et al., 2015; O'Gorman et al., 2016), and (ii) structured paradigms which represent events by marking text spans and adding additional arguments (e.g., participants, location etc.) to create a structured template (Grishman and Sundheim, 1996). Event categorization divides extraction paradigms into two types: (i) ontology-driven paradigms that are limited to specific event types (Grishman and Sundheim, 1996; Doddington et al., 2004), and (ii)

ontology-free paradigms that do not place type restrictions (Pustejovsky et al., 2003b; Araki and Mitamura, 2018).

Throughout this chapter, we use a sentence-level, span-based, ontology-free event extraction paradigm. Sentence-level extraction suits our domains of interest since literary texts, clinical notes and clinical conversations often discuss multiple events per sentence. Span-based and ontology-free extraction allows us to develop coding guidelines that are easily adaptable across domains, since event arguments and types are usually domain-specific or task-specific. This adaptability sets our work apart from other prior work on medical event extraction such as adverse drug event extraction (Nikfarjam et al., 2015; Sarker and Gonzalez, 2015; Cocos et al., 2017; Henry et al., 2020) and personal event extraction from online support groups (Wen et al., 2013; Naik et al., 2017), which focus on specific event types. Our guidelines draw heavily from the Thyme-TimeML guidelines (Styler IV et al., 2014) used by the Clinical TempEval challenges on event ordering in clinical notes (Bethard et al., 2015, 2016, 2017),[1] but also cover event extraction in a novel non-narrative domain: doctor-patient conversations.

3.2.2 Unsupervised Domain Adaptation Techniques

As discussed in the previous chapter, unsupervised domain adaptation techniques aim to transfer a model from a source domain to a target domain, using only unlabeled data from the target domain. Typically, most methods achieve this by learning some form of alignment between the marginal distributions of source and target domains. This section provides a concise summary of popularly used unsupervised adaptation methods from each coarse category in our adaptation method taxonomy to contextualize our choice of baselines; for a more comprehensive overview, we refer interested readers to Ramponi and Plank (2020).

Model-centric techniques: Early work primarily focused on developing feature-centric approaches such as structural correspondence learning (SCL) (Blitzer et al., 2006, 2007), which tried to perform unsupervised adaptation by mapping source and target examples into a shared **pivot feature** space. Here pivot features are selected to be features that behave the same way for discriminative learning in both domains (e.g., sentiment terms such as *amazing* and *great* show similar behavior for sentiment analysis across domains). Rapid advances in neural representation learning further pushed the development of feature-centric approaches, including neural variants of SCL (Ziser and Reichart, 2017) and autoencoder-based methods (Glorot et al., 2011; Chen et al., 2014). The rise of neural models also led to a surge in the development of loss-centric approaches, largely spearheaded by work on adversarial domain adaptation, which tries to learn domain-agnostic representations useful for the task of interest (Ganin and Lempitsky, 2015; Ganin et al., 2016). Both feature-centric and loss-centric approaches have shown promising performance on sequence labeling (Gui et al., 2017; Xing et al., 2018; Naik and Rose, 2020), with the case study from the previous chapter also establishing the dominance of loss-centric approaches for event extraction

[1] We provide a detailed comparison with this work in §3.3.2.

from clinical narratives. On the other hand, unsupervised adaptation methods from parameter-centric or ensemble categories have not been explored as much, especially for sequence labeling. Wright and Augenstein (2020) demonstrated the utility of ensemble adaptation methods such as mixture-of-experts for text classification tasks, but in a multi-source setting (i.e., multiple source datasets are available). Parameter-centric methods such as initializing model weights using the weights of a model trained on a different but related task have shown some success on text classification (Gee and Wang, 2018; Vlad et al., 2019). Since we are considering a single-source unsupervised setting and interested in the task of event extraction from various clinical texts, we choose adversarial domain adaptation (ADA), a loss-centric method as a strong baseline from the model-centric category.

Data-centric techniques: From the data-centric category, the fine model categories explored most in an unsupervised setting are pretraining and pseudo-labeling. Pretraining techniques like domain adaptive finetuning (DAFT), which learns a joint language model for both source and target domain text, have shown great success for unsupervised sequence labeling (Han and Eisenstein, 2019; Caselli et al., 2021). Additionally, pseudo-labeling methods like self-training have shown some success (Naik and Rose, 2020), but our experiments from the previous chapter demonstrated that these methods don't deal well with event/entity spans containing highly technical vocabulary. Other fine model categories such as noising/denoising and instance learning have not shown much promise in an unsupervised setting. For example, Tourille et al. (2017) demonstrate that a noising strategy that replaces some event tokens with "UNK" tokens does not improve performance on event extraction. The last fine category, active learning, does not lend itself to an unsupervised setting since it involves acquiring annotations for small subsets of target domain data. Under these considerations, we choose domain adaptive finetuning (DAFT), a pretraining method as a strong data-centric baseline.

Hybrid techniques: Hybrid techniques are the least-explored coarse category of methods in an unsupervised adaptation setting since most methods expect a small proportion of labeled data from the target domain (Jiang and Zhai, 2007). Some classifier-based instance weighting methods that train a classifier to discriminate between source and target domain instances, and then compute source instance weights using this classifier can be used in an unsupervised setting. However, they have shown mixed to negative results across various tasks including sequence labeling (Søgaard and Haulrich, 2011; Plank and Moschitti, 2013; Plank et al., 2014). We try to advance research on this class of methods by proposing a new unsupervised instance weighting technique called likelihood-based instance weighting (LIW), which leverages the power of language models for instance weight computation.

3.2.3 Active Learning Techniques

Active learning is a fairly popular learning paradigm in the field of machine learning in which the learning algorithm is allowed to "be curious", i.e., allowed to choose data to learn from, in

the hope that it will achieve better performance with less training (Settles, 2009). Active learning techniques select unlabeled data instances and *query* an *oracle* (e.g., a human annotator) to obtain labels for these instances. By strategically choosing informative instances to query, active learning techniques attempt to minimize the cost of obtaining labeled data. Prior literature has explored three different settings for choosing instances to learn from: (i) membership query synthesis, (ii) stream-based sampling, and (iii) pool-based sampling. In membership query synthesis, the learner can generate instances instead of just sampling from a data distribution. In stream-based learning, unlabeled instances are obtained one at a time from a distribution or source and a learner must decide to keep or discard them. Finally, in pool-based sampling, the learner has access to a large pool of unlabeled instances and can choose a batch of *most informative* instances to learn from at each iteration. Pool-based sampling is the most commonly explored setting since it is typical to have large sets of unlabeled data available, and our work also uses this setting for the same reason.

Using active learning can be desirable when dealing with complex tasks in high-expertise domains, for which sourcing annotations is extremely difficult, time-consuming, and expensive. Their property of boosting data efficiency makes active learning methods an interesting avenue to explore in a domain adaptation scenario, in which we can use them to identify (and label) the most informative subset of data from a target domain of interest and adapt to the domain better. We briefly summarize prior work on incorporating active learning during adaptation; for comprehensive general overviews of active learning methods in machine learning and natural language processing, we refer interested readers to Settles (2009); Olsson (2009); Schröder and Niekler (2020).

The seminal work by Rai et al. (2010) proposed two ways of incorporating active learning during adaptation. First, instead of training a model on target domain data from scratch, they trained it on source domain data, followed by further training on selected target domain data. This strategy results in a better starter model for active learning. Second, they trained a domain discriminator on the task of distinguishing source and target domain instances, and used this classifier to filter out target instances that were extremely similar to the source domain since a source-trained model should already be able to label them accurately. Their experiments showed promising results on the task of sentiment analysis. Follow-up work on sentiment analysis explored other strategies such as training separate models on source and target domain data and using them in a query-by-committee strategy (Li et al., 2013), or adapting general-domain sentiment lexicons to a target domain to train a strong starter model (Wu et al., 2017). Aside from sentiment analysis, active learning has been incorporated into domain adaptation for word sense disambiguation by learning stronger priors before sampling (Chan and Ng, 2007). Some work has also looked at a reverse setting, in which a model is first trained on target domain data and then used to select source domain instances in an active learning loop (Shi et al., 2008). Aside from NLP tasks, the idea of combining active learning and domain adaptation, particularly leveraging domain discriminators, has also been explored in computer vision (Su et al., 2020; Fu et al., 2021; Xie et al., 2021).

Similar to Rai et al. (2010), we train a better starter model on source domain data, and try to incorporate *domain-awareness* while sampling instances. However, instead of using a domain

discriminator to filter instances, we compute a probability odds ratio of an instance belonging to the target domain and use it to weight instances. We also design a domain-awareness criterion based on embedding space similarity, and compare the performance of both strategies.

3.3 Creating Event Extraction Datasets for Additional Domains

In this chapter, we extend our space of domains of interest by bringing two additional domains (aside from clinical narratives) under our purview: (i) literary texts, and (ii) doctor-patient conversation transcripts. As in the previous chapter, we continue to use news articles as our source domain. The literary text domain is also a high-expertise narrative domain like the clinical narrative domains explored in the previous chapter, while the doctor-patient conversation domain falls under both high-expertise and non-narrative categories. For literary text, we use the event extraction dataset developed by Sims et al. (2019). For the remaining domains (clinical notes and conversations), we develop new event extraction datasets using the procedure detailed in this section. We first collect raw documents for both domains, followed by sampling documents from specific clinical specialties to control for topical variation. Then we carefully construct event annotation guidelines for our data domains by adapting the existing TimeML scheme (Pustejovsky et al., 2003a). Finally we conduct the annotation process using these guidelines and construct test datasets for both domains.

3.3.1 Document Collection for Clinical Domains

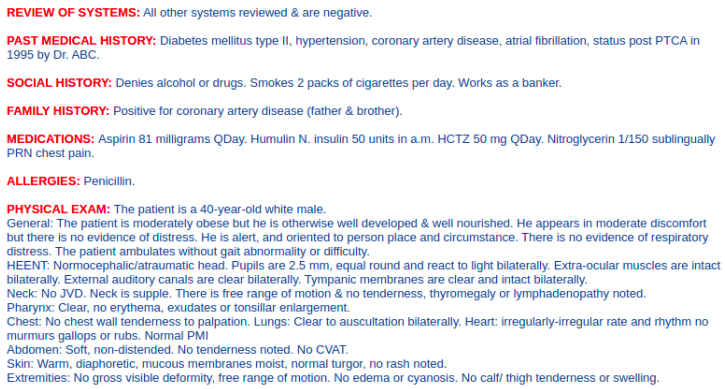
Clinical Notes

Clinical notes are records documenting physician observations from their interactions with patients. They usually detail various aspects of a patient's care such as present illness, symptoms, medical history, treatments, and test results. They share a thematic structure, though particular specialties (e.g., cardiology) and institutions often incorporate their own modifications. We collected a set of 4999 de-identified clinical notes from 40 specialties, by scraping mtsamples.[2] Average length of a clinical note in this dataset is 652 tokens. Figure 3.1 shows a sample clinical note from our dataset.

Doctor-Patient Conversations

This dataset contains human-transcribed, de-identified conversations recorded during physician-patient visits. The conversations often follow a similar schema, with patients describing their symptoms, doctors inquiring about ongoing treatments, and then suggesting potential follow-up treatments/tests. We used a dataset of 63,540 conversation transcripts covering 53 specialties, sampled from Verilogue Inc.'s proprietary database of in-office medical conversations. Verilogue is an ethnographic market research organization that contracts with physicians across

[2] https://www.mtsamples.com/



REVIEW OF SYSTEMS: All other systems reviewed & are negative.

PAST MEDICAL HISTORY: Diabetes mellitus type II, hypertension, coronary artery disease, atrial fibrillation, status post PTCA in 1995 by Dr. ABC.

SOCIAL HISTORY: Denies alcohol or drugs. Smokes 2 packs of cigarettes per day. Works as a banker.

FAMILY HISTORY: Positive for coronary artery disease (father & brother).

MEDICATIONS: Aspirin 81 milligrams QDay. Humulin N. insulin 50 units in a.m. HCTZ 50 mg QDay. Nitroglycerin 1/150 sublingually PRN chest pain.

ALLERGIES: Penicillin.

PHYSICAL EXAM: The patient is a 40-year-old white male.
General: The patient is moderately obese but he is otherwise well developed & well nourished. He appears in moderate discomfort but there is no evidence of distress. He is alert, and oriented to person place and circumstance. There is no evidence of respiratory distress. The patient ambulates without gait abnormality or difficulty.
HEENT: Normocephalic/atraumatic head. Pupils are 2.5 mm, equal round and react to light bilaterally. Extra-ocular muscles are intact bilaterally. External auditory canals are clear bilaterally. Tympanic membranes are clear and intact bilaterally.
Neck: No JVD. Neck is supple. There is free range of motion & no tenderness, thyromegaly or lymphadenopathy noted.
Pharynx: Clear, no erythema, exudates or tonsillar enlargement.
Chest: No chest wall tenderness to palpation. Lungs: Clear to auscultation bilaterally. Heart: irregularly-irregular rate and rhythm no murmurs gallops or rubs. Normal PMI
Abdomen: Soft, non-distended. No tenderness noted. No CVAT.
Skin: Warm, diaphoretic, mucous membranes moist, normal turgor, no rash noted.
Extremities: No gross visible deformity, free range of motion. No edema or cyanosis. No calf/ thigh tenderness or swelling.

Figure 3.1: Sample clinical note from mtsamples.com.

```
DR: All right, so um, when did you start showing symptoms?
PT: Um, maybe about three years ago.
DR: No, no, these current symptoms. Not your COPD, I mean -
PT: Oh, you mean the fever?
DR: Yes, sorry, the fever, when did you -
PT: Um, the last five days.
DR: Oh, okay.
PT: I went flying out to the lake, I was just concerned that -
DR: Yeah.
PT: I have been sneezing and coughing a lot but -
DR: Um-hum.
PT: Since the fever started, I was concerned that it might be something serious.
DR: Okay. Are you using your nasal medicine?
PT: Yes.
DR: Okay. Regularly?
PT: Yes.
```

Figure 3.2: Sample snippet from a physician-patient conversation transcript.

a variety of specialties to record natural in-office conversations with their patients who agree to participate in the research by providing verbal and written consent. Recordings are made on a digital recording device or a smartphone application and are uploaded to a secure server where they are scrubbed of all identifiable information, in accordance with the Health Insurance Portability and Accountability Act of 1996 (HIPAA) privacy rule. De-identified recordings are transcribed and stored in Verilogue's database, which currently contains over 100,000 recordings dating from 2006 to 2017. Average conversation transcript length in this dataset is 2309 tokens. Figure 3.2 shows a snippet from a conversation transcript. Note that since this dataset is not publicly shareable, this is a constructed snippet and not an actual sample from our data.

Linguistic Differences between Domains

Both clinical domains chosen for dataset construction exhibit different types of linguistic shifts from the source domain (news). While both domains exhibit a shift in vocabulary, this shift is more pronounced in clinical notes since they are written by doctors (experts) who use highly technical

Specialty	#Notes	#Convos
Cardio	372	4876
Obgyn	160	1784
Onco	90	7177

Table 3.1: Domain-wise raw data statistics for chosen medical specialties.

terms. Conversely, shifts in syntax are more pronounced in conversations due to the prevalence of repetition, back-channeling, interruptions etc. Semantic shifts are more pronounced in conversations since they contain a higher proportion of hypothetical statements (e.g., when doctors ask questions, make requests or "think out loud") than both notes and news articles which tend to serve as records of actual events. To better evaluate model performance on linguistic shifts, we control for topical variation across domains by limiting our focus to 3 specialties: Cardiovascular/Pulmonary (Cardio), Obstetrics/Gynaecology (Obgyn) and Hematology/Oncology (Onco). These specialties are well-represented in both notes and conversations, and cover a variety of event types ranging from intervals with fixed duration (e.g., pregnancy), to intervals with indeterminable endpoints (e.g., long-term cardiac failure). Table 3.1 gives an overview of the number of notes and conversations in each chosen specialty.

3.3.2 Developing Event Annotation Guidelines

We develop a set of coding guidelines for the task of annotating event triggers in documents collected from these two clinical domains. Our coding guidelines build upon TimeML (Pustejovsky et al., 2003a), a rich specification language for annotation of events and temporal expressions in text,[3] and Thyme-TimeML (Styler IV et al., 2014), a variant of TimeML developed for clinical notes. We start with these guidelines because they use a syntax-driven domain-agnostic definition of events, allowing for an adaptable annotation scheme. In TimeML, the term *event* refers to situations that *happen* or *occur*, or circumstances in which something *obtains* or *holds true*. This is a broad definition, consistent with Bach's definition of **eventualities** (Bach, 1986), and the idea of **fluents** (McCarthy, 2002). Events can be expressed in text by means of tensed or untensed verbs, nominalizations, adjectives, predicative clauses or prepositional phrases. TimeML describes rules to annotate events in all these syntactic categories. Styler IV et al. (2014) adapted these rules for clinical notes. They focused on the THYME corpus of 1254 de-identified notes from the Mayo Clinic, representing two fields in oncology: brain cancer and colon cancer. As a first step, we annotate one document from each of our domains following TimeML and Thyme-TimeML rules. During this phase, we identify cases where it is reasonable to deviate from these guidelines.

[3]The complete TimeML coding manual is available here: https://catalog.ldc.upenn.edu/docs/LDC2006T08/timeml_annguide_1.2.1.pdf

Deviations from TimeML

Our guidelines differ from TimeML in their treatment of two categories:

- **Activity patterns:** Activity patterns are events that are neither pure generics[4], nor single events clearly positioned in time. For example, consider the sentence "I *take* my blood pressure regularly." The event *take* is not grounded in time. It is also not a pure generic event as it is definitely associated with the speaker. Such events are *not* annotated in TimeML. However, in our data, these activity patterns occur frequently in crucial contexts such as taking medications, following lifestyle changes suggested by doctors, measuring vital signs, etc.

- **Long-term states:** Because TimeML was geared towards the task of temporal ordering, it strictly restricted annotation of stative events to the following types: (i) states associated with a temporal expression, (ii) states undergoing a change within the document, (iii) states introduced by other events, since those can offer temporal cues, and (iv) states associated with the document creation time. However, many stative events in our data don't fit within these strict parameters, but are nevertheless important. The most crucial category is states associated with long-term ongoing illnesses (e.g., "The patient has a long history of *COPD*").

These event categories are not specific to clinical domains only. For example, long-term state events might be salient when extracting personal events from biographies (e.g., "Bill Gates is currently *employed* full-time at the Bill and Melinda Gates Foundation."). Similarly activity patterns might be salient when extracting events from scientific procedure manuals ("*Repeat* step 5 daily, over a period of 30 days."). Considering the general utility of these event categories, we add rules to extract these two categories of events. We also expand syntactic rules to cover constructions unique to doctor-patient conversations such as repetition, especially for instructions, and hypothetical event annotation in utterances when doctors are "thinking out loud".

Deviations from Thyme-TimeML

Our guidelines differ from Thyme-TimeML in their treatment of two categories:

- **Generic events:** Thyme-TimeML annotates generic events present in sections documenting doctors' discussion of risks, plans and alternative strategies. They do so because adding these events to a patient's clinical timeline could be important from a legal perspective, as they help to establish informed consent and knowledge of risk. We do not annotate pure generics, because we do not perceive any domain-agnostic utility in annotating them. Note that we annotate verbs of discussion and comprehension which are not generics, so we do not completely ignore events associated with patient consent. For example, in the sentence

[4]Pure generics are events which discuss illnesses/treatments in general, and are not associated with a specific person and time. For example, "there is a *benefit* to systemic adjuvant *chemotherapy*."

Domain	Entity κ	Event κ
Notes	0.9117	0.8652
Convos	0.8634	0.8327

Table 3.2: Inter-annotator agreement on entity and event annotation tasks in both domains, measured using chance-corrected Cohen's κ.

"She repeated the potential side effects back to me," *repeated* is annotated, but *effects* is not. Thyme-TimeML would have annotated both.

- **Entities as events:** Thyme-TimeML treats some entities and non-events as events in clinical language. Two categories see this shift in semantic interpretation: (i) Medications, and (ii) Disorders. Both categories contribute significant information to a patient's timeline, and so they are treated as events. Since we are not specifically focused on timeline construction, we do not treat these as events. To ensure that we do not discard potentially crucial information, we incorporate an additional step in which we annotate entities such as medications, body parts, abnormalities (e.g., rash), etc.

Example Annotations

Following are some example sentences from both clinical domains, annotated with events according to our coding guidelines:

1. **Clinical note snippets:**
   (a) Sample Name: *Excision* of Squamous Cell *Carcinoma*.
   (b) *Re-excision* of squamous cell carcinoma site, right hand
   (c) The tissue was *passed* off the field as a specimen
   (d) Cardiovascular system review: Chest *pain* in retrosternal area
2. **Conversational utterances:**
   (a) I have been *taking* Midol for 6 months.
   (b) Your *surgery* was *done* last year, was it?
   (c) Your leg is a little more *swollen*.
   (d) Do you *take* your blood *pressure* daily?

These sentences contain events that fall into various syntactic categories (nouns, verbs, adjectives, etc.). Sentences 1a and 1d contain examples of long-term state events (*carcinoma* and *pain*), while sentences 2a and 2d contain examples of activity patterns (*taking* and *take*).

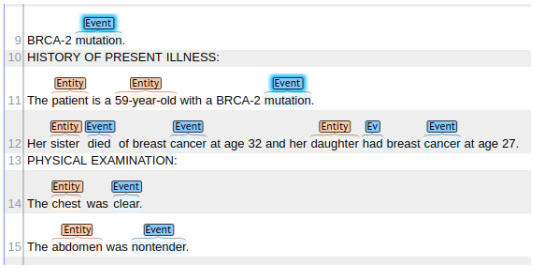


Figure 3.3: Sample clinical note with entity and event annotation.

Statistic	News	Notes	Convos
#Files	54	45	45
#Tokens	18,263	28,935	76,711
#Events	1986	4781	7064
Event Density	10.88%	16.52%	9.21%
Vocab Size	3978	4303	3505
Event Vocab	1015	1588	1472

Table 3.3: Dataset statistics. Note that the statistics for TimeBank (News) are computed over the test set for fair comparison with our datasets, which are test-only.

3.3.3 Annotation Process

After creating our guidelines[5], we validate them by having two expert annotators annotate one document from each domain. We observe high inter-annotator agreement (measured by chance-corrected Cohen's κ) on both entity and event annotation, in both clinical domains. Table 3.2 presents the agreement scores. To create our final datasets, we sample 45 documents from each domain (15 from each specialty). Each document is annotated by one expert. Annotation is carried out using the BRAT stand-off markup interface (Stenetorp et al., 2012). Figure 3.3 shows a sample clinical note annotated with events and entities. Table 3.3 gives a brief overview of statistics for our datasets, in comparison with the TimeBank dataset of news articles (Pustejovsky et al., 2003b). Note that Appendix C presents additional examples of annotated instances from all these datasets.

3.4 Case Study I: Evaluating LIW on Unsupervised Adaptation

In this study, we focus on building adaptable event extraction models that work well for our chosen clinical domains without using any in-domain annotated training data, since collecting annotated data is often expensive and time-consuming, especially for high-expertise domains like medicine. To achieve this, we exploit the availability of annotated training data in the news domain, and explore the possibility of adapting models trained on news to these clinical domains in an

[5]Complete coding guidelines are available in appendix A

unsupervised manner. As described in §3.3.1, both clinical domains exhibit shifts from the news domain at various linguistic levels. This case study tries to specifically address shifts in surface realization (i.e., shifts at lexical and syntactic levels), by tackling the problem of misalignment between the marginal distributions of the source and target domains.

The marginal misalignment problem is analogous to the problem of covariate shift in transfer learning. Covariate shift arises when the marginal distribution (or input distribution) $P(X)$ changes between train (source) and test (target) data. Therefore, directly applying a supervised model trained on the source set, to the target data does not perform well due to the gap between source and target marginal distributions (Shimodaira, 2000). To handle this issue by better aligning marginal distributions of both domains, and advance work on unsupervised hybrid methods, we propose a new instance weighting technique: likelihood-based instance weighting (LIW).

3.4.1 Likelihood-based Instance Weighting

Data selection and instance weighting strategies have frequently been used to perform supervised domain adaptation by correcting for distributional shifts (Jiang and Zhai, 2007; Foster et al., 2010; Axelrod et al., 2011; Wang et al., 2017). As a reminder, the underlying premise behind these techniques is that some instances from target data and source data often share certain characteristics. Training only on these similar instances (by pruning out other dissimilar instances), or biasing training to focus more on these similar instances (by weighting) can produce models that perform better on target data. Motivated by this, we design an instance weighting strategy that uses likelihood scores computed by a language model to weight instances. The instance weight computation in LIW works as follows.

Let $S_t = w_1 w_2 ... w_n$ be a sentence from the source training set. Let O be a language model trained on raw text from the target domain. We first compute the likelihood of sentence S_t under O as $\mathbb{L}_t = P_O(w_1)\Pi_{i=2}^{n} P_O(w_i|w_1 ... w_{i-1})$, where P_O indicates probability under model O. Then we compute a weight for S_t as follows:

$$\alpha_{S_t} = \frac{\mathbb{L}_t}{\sum_{i=1}^{|N|} \mathbb{L}_i} * |N| \tag{3.1}$$

where $|N|$ is the size of in-domain training set. This metric gives a higher weight to source sentences that are *more likely* under the target domain language model, up-weighting instances that share more characteristics, and are consequently better aligned with target domain sentences. The alpha values are used to weight the loss function, thus biasing the training procedure to focus more on these better-aligned sentences. Doing so improves alignment between the marginal distributions of source and target domains. From a linguistic perspective, we also hope that using language model scores to reweight instances helps this technique leverage word contexts in addition to relying on vocabulary shared between source and target domains, like most methods studied in the previous chapter.

3.4.2 Baseline Adaptation Methods

In addition to evaluating the performance of our proposed method (LIW), we also contrast its performance with strong unsupervised adaptation baselines from the remaining two coarse categories. From the model-centric category of methods, we choose adversarial domain adaptation (ADA), a loss augmentation method that achieved the best performance on extracting events from clinical narratives in our case study from the previous chapter. From the data-centric category of methods, we choose domain adaptive finetuning (DAFT), a pretraining method that showed promising results on unsupervised sequence labeling tasks (Han and Eisenstein, 2019), in place of the continuous pretraining strategy (Gururangan et al., 2020) that did not perform well in our previous case study. Note that we do not choose a pseudo-labeling baseline because our event spans are likely to contain medical vocabulary, and our analysis of strengths and weaknesses from the previous chapter showed that pseudo-labeling methods did not deal well with such spans. Another interesting distinction to note is that ADA is a task-guided technique since it performs task training and adaptation jointly, while DAFT and LIW are task-agnostic techniques since they perform adaptation first, followed by task training.

Adversarial Domain Adaptation

Adversarial domain adaptation was first proposed by Ganin and Lempitsky (2015), who showed its efficacy on sentiment analysis, among other machine learning tasks. It is an unsupervised domain adaptation technique, inspired by theory on domain adaptation which suggests that effective domain transfer can be achieved when model predictions are based on features that cannot discriminate between source and target domains. It operationalizes this theory in a representation learning approach which promotes the emergence of features that *are discriminative* for the main task on the source domain (event extraction in our case), but *not discriminative* with respect to shifts between source and target domains. In addition to sentiment analysis, adversarial domain adaptation (ADA) has been successfully applied to other NLP tasks such as duplicate question detection, part-of-speech tagging and answer retrieval for question answering (Ganin et al., 2016; Li et al., 2017; Liu et al., 2017; Gui et al., 2017; Chen et al., 2018b; Shah et al., 2018; Yu et al., 2018).

We describe how we adapt ADA for our task of event extraction (Naik and Rosé, 2020). Figure 3.4 gives an overview of our ADA framework for event extraction. It consists of three components: i) representation learner (R) ii) event classifier (E) and iii) domain predictor (D). The representation learner generates token-level representations, while the event classifier and domain predictor use these representations to identify events and predict the domain to which the sequence belongs. The key idea is to train the representation learner to generate representations which *are predictive* for event classification but *not predictive* for domain prediction, introducing domain-invariance which makes it more robust to shifts between source and target domains. A notable benefit of ADA is that the only data we need from the target domain is unlabeled data.

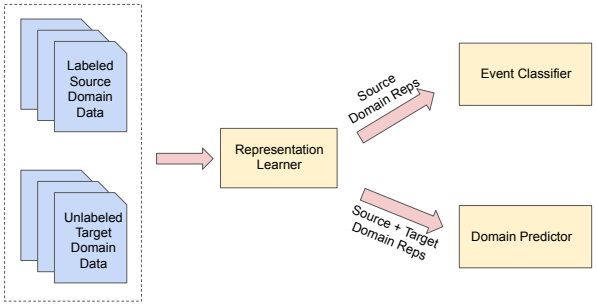


Figure 3.4: Adversarial domain adaptation framework for event trigger identification.

To ensure domain-invariance in representation learning, ADA uses adversarial training which works as follows. Assume that we have a labeled source domain dataset D^s with examples $\{(x_1^s, e_1^s), ..., (x_n^s, e_n^s)\}$, where x_i^s is the token sequence and e_i^s is the sequence of event tags. We construct an auxiliary dataset D^a with examples $\{(x_1^a, d_1^a), ..., (x_n^a, d_n^a)\}$, where x_i^a is the token sequence and d_i^a is the domain label, using token sequences from D^s and unlabeled target domain sentences. The representation learner R maps a token sequence $x_i = (x_{i1}, ..., x_{ik})$ into token representations $h_i = (h_{i1}, ..., h_{ik})$. The event classifier E maps representations $h_i = (h_{i1}, ..., h_{ik})$ to event tags $e_i = (e_{i1}, ..., e_{ik})$. The domain predictor D creates a pooled representation $p_i = Pool(h_{i1}, ..., h_{ik})$ and maps it to domain label d_i^a. Given this setup, we apply an alternating optimization procedure. In the first step, we train the domain predictor using D^a, to optimize the following loss:

$$\arg\min_{D} \mathcal{L}(D(h_i^a), d_i^a)$$

In the second step, we train the representation learner and event classifier using D^s to optimize the following loss:

$$\arg\min_{R,E} \Big[\sum_k \left(\mathcal{L}(E(h_{ik}^s), e_{ik}^s) \right) - \lambda \mathcal{L}(D(h_i^s), d_i^s) \Big]$$

$\mathcal{L}$ refers to the cross-entropy loss and λ is a hyperparameter. In practice, the optimization in the above equation is performed using a gradient reversal layer (GRL) (Ganin and Lempitsky, 2015). A GRL works as follows. During the forward pass, it acts as the identity, but during the backward pass it scales the gradients flowing through by $-\lambda$. We apply a GRL g_λ before mapping the pooled representation to a domain label using D. This changes the optimization to:

$$\arg\min_{R,E} \Big[\mathcal{L}(D(g_\lambda(p_i^s)), d_i^s) + \textstyle\sum_k \mathcal{L}(E(h_{ik}^s), e_{ik}^s) \Big]$$

Particular implementation details such as the network architectures used for R, E and D are

described in detail in §3.4.3.

Domain Adaptive Fine-tuning

Domain adaptive fine-tuning (DAFT) has recently been proposed as an effective pretraining technique for unsupervised adaptation of sequence labeling models that use contextualized embeddings (Han and Eisenstein, 2019). This technique was proposed to tackle scenarios in which a sequence labeling model is trained on a canonical source domain (e.g., news), and applied to a different target domain. Additionally, the sequence labeling model uses contextualized embeddings, such as BERT, that have been pretrained on a corpus distinct from both source and target domains. DAFT addresses these shifts by leveraging the power of masked language modeling on source and target domain texts to improve alignment between marginal distributions of source and target domains. This technique has been shown to be extremely effective for unsupervised transfer, even to challenging domains such as Early Modern English and social media (Han and Eisenstein, 2019). The DAFT procedure works as follows:

1. Create a large dataset containing equal proportions of sentences from source and target domains. Fine-tune contextualized embeddings using a masked language modeling objective.

2. Using fine-tuned embeddings, train an event extraction model on labeled source data.

In addition to this setup, we experiment with a variant of this procedure, which uses a syntactic objective function in place of masked language modeling. This variant fine-tunes embeddings on the task of predicting part-of-speech tags in step 1. The motivation behind this variant is two-fold. First, we observe that event annotation is heavily syntax-driven, allowing delexicalized models (i.e., models using POS tags instead of words) to achieve high performance (§3.4.3). This indicates that infusing additional syntactic awareness into embeddings might help performance on the task. Second, syntax can offer an additional basis for alignment, since sentences that look very different lexically, might follow similar syntactic structures. Intuitively, this variant is similar to syntactic relexicalization which has shown success in cross-lingual dependency parsing (Duong et al., 2015).

Complete model architecture details for all techniques are described in §3.4.3.

3.4.3 Experiments

The goal of our evaluation is two-fold: (i) evaluate the efficacy of our proposed technique (LIW) for unsupervised adaptation of event extraction, and (ii) analyze which adaptation techniques work best for each clinical domain of interest and try to identify aspects of the domains (or datasets) that make these techniques work. For our baseline task model, we choose a strong BERT-BiLSTM model with no transfer. We then evaluate the performance of all adaptation techniques by applying them to this baseline model. All techniques are evaluated in a zero-shot setting, wherein models are trained on TimeBank (Pustejovsky et al., 2003a) and tested on one of our clinical datasets (clinical notes or doctor-patient conversations).

Model Details

We evaluate the performance of the following models on our datasets:

- **VERB:** A simple unsupervised baseline labeling all verbs as events.
- **DELEX:** A fully-delexicalized baseline using POS tag embeddings as features, followed by an MLP (multi-layer perceptron).
- **BERT:** A single-layer BiLSTM over contextual embeddings extracted using BERT (Devlin et al., 2019), followed by an MLP, similar to the best-performing model on LitBank (Sims et al., 2019).
- **CBERT:** Similar to BERT, but embeddings are extracted from Clinical-BERT (Alsentzer et al., 2019).
- **BERT-ADA:** BERT baseline trained using adversarial domain adaptation. The domain predictor adversary is an MLP classifier which uses max pooling to compute the pooled representation.
- **BERT-LIW:** BERT baseline trained on data weighted by LM likelihood. We train autoregressive language models using 3 million tokens for each target domain.
- **BERT-DAFT:** BERT baseline with domain adaptive fine-tuning. We use the same target domain text used to train LMs for BERT-LIW. For news, we extract 3 million tokens from the CNN/ DailyMail dataset (Hermann et al., 2015).
- **BERT-DAFT-SYN:** BERT baseline with syntactic fine-tuning on the same source+target text as BERT-DAFT, POS tagged using Stanford CoreNLP (Manning et al., 2014).

Overall Performance

Tables 3.4 and 3.5 show the performance of all models when transferring from news (in-domain) to clinical notes and doctor-patient conversations (out-of-domain) respectively. From the tables, we see that the DELEX baseline is surprisingly strong out-of-domain, reaching nearly 60.4 F1 on conversations. BERT with no transfer performs well out-of-domain, improving by 8.25 F1 points on average over DELEX. C-BERT also performs well out-of-domain, but does worse than the vanilla BERT baseline. We hypothesize that this could be attributed to the fact that fine-tuning only on clinical notes does not improve alignment between source and target domains, providing no basis for models trained on news to adapt better. This echoes our observations from the previous case study, where we saw that continuous pretraining was not a very effective adaptation technique in an unsupervised setting. BERT-ADA, the loss-centric baseline, shows mixed results, improving over BERT by 2.4 F1 on notes, but dropping by 1.1 F1 on conversations. BERT-LIW, our method, and BERT-DAFT, the pretraining baseline, improve upon BERT in both settings. BERT-DAFT shows

Model	In-Domain			Out-of-Domain		
	P	R	F1	P	R	F1
VERB	58.8	66.5	62.5	49.4	41.4	45.0
DELEX	75.0	66.3	70.4	74.4	42.2	53.8
BERT	80.6	86.0	83.2	85.7	55.9	67.6
CBERT	79.2	83.3	81.2	85.8	52.9	65.4
BERT-ADA	81.2	86.3	83.7	83.2	**60.4**	**70.0**
BERT-LIW	81.9	86.6	84.1	**86.7**	56.0	68.1
BERT-DAFT	79.1	85.9	82.3	83.9	58.6	69.0
BERT-DAFT-SYN	76.9	80.7	78.7	70.7	56.8	63.0

Table 3.4: Model performance on unsupervised domain transfer experiments from news to clinical notes.

Model	In-Domain			Out-of-Domain		
	P	R	F1	P	R	F1
VERB	58.8	66.5	62.5	44.6	68.1	53.9
DELEX	75.0	66.3	70.4	56.9	64.5	60.4
BERT	80.6	86.0	83.2	75.0	63.6	68.9
CBERT	79.2	83.3	81.2	66.5	65.1	65.8
BERT-ADA	81.1	85.9	83.4	**74.5**	62.2	67.8
BERT-LIW	80.0	87.0	83.4	72.8	67.3	70.0
BERT-DAFT	78.5	84.8	81.5	72.7	**73.1**	**72.9**
BERT-DAFT-SYN	80.0	78.7	79.3	67.6	60.7	63.9

Table 3.5: Model performance on unsupervised domain transfer experiments from news to doctor-patient conversations.

minor performance drops in-domain, possibly due to some degree of catastrophic forgetting. BERT-DAFT-SYN shows performance drops, both in-domain and out-of-domain, in both settings. Unlike syntactic relexicalization work which used non-contextualized embeddings, we use contextualized embeddings, which already possess a larger degree of syntactic information, probably reducing the need for syntax-driven training. Another source of errors is automatic part-of-speech tagging, since off-the-shelf taggers trained on news will be less accurate on our data. Across domains, the skew between precision and recall is higher on notes, which might stem from the specialized vocabulary used in them dragging down recall. Overall, most adaptation methods seem to help improve event extraction performance on both clinical domains, with our best models achieving F1 scores of 70.0 and 72.9 on notes and conversations respectively with *no* training data.

3.4.4 Analysis and Discussion

Tables 3.4 and 3.5 provide some indication of the ability of different adaptation techniques to handle shifts between source and target domains. However, this overall evaluation does not account for shifts at different linguistic levels. As we state earlier, domain shifts can occur at multiple layers in language (e.g., lexical level, syntactic level, etc.), leading to different dimensions of variation between domains (e.g., topical variation, genre variation, etc.). To probe this, we perform a deeper analysis of model performance, focusing on two questions:

1. How well do models handle lexical shifts between domains?
2. How much does performance differ between examples that do/do not exhibit semantic shifts?

To answer the first question, we hone in on model performance under lexical shift by evaluating performance on out-of-vocabulary (OOV) cases. For the second question, we use event type as a proxy to distinguish between target domain events that demonstrate semantic shifts from source domain events, and target domain events that do not. This proxy is motivated by our observation that the new event types we add (activity patterns and long-term states) are unique to the target domains and often require an understanding of the event beyond its textual manifestation. For example, consider the sentence "Taking Midol for period pain is recommended". Based purely on textual content, *taking* would be considered a pure generic and not an event. However, if we include the additional context that this is a statement offered as direct advice by a doctor to a patient, or written under medications in a patient's clinical note, *taken* becomes an activity pattern because it is now associated with an implicit participant (the patient). So we annotate a random sample of 500 events from each of our test datasets with event types and evaluate performance separately on event types present in source vs those absent in source.

Performance on OOV Cases

To answer the question of lexical shift, we separate model performance on in-vocabulary (IV) and out-of-vocabulary (OOV) tokens. Note that the proportion of events that are OOV is higher in clinical notes (52%) than conversations (20.6%). Tables 3.6 and 3.7 present model performance on these token categories. We observe that all models fare reasonably well on OOV tokens, however there is still a large gap between F1 scores on IV and OOV tokens. Performance trends on OOV tokens are similar to trends on the full dataset, with LIW and DAFT showing improvement on both domains and ADA showing improvement primarily on clinical notes. Surprisingly, despite the use of specialized language, OOV performance on clinical notes is higher than conversations for all models except BERT-DAFT. The lower performance of BERT-DAFT on notes is consistent with our observation from the previous case study, where we saw that for clinical narratives, loss augmentation methods like ADA did better at handling technical vocabulary than pretraining methods like DAFT. Taking a closer look at the OOV event instances from clinical notes that

Model	IV F1	OOV F1
BERT	73.5	61.2
BERT-ADA	75.2	65.0
BERT-LIW	73.6	62.6
BERT-DAFT	75.7	62.0
BERT-DAFT-SYN	67.7	58.4

Table 3.6: Model performance on in-vocabulary (IV) and out-of-vocabulary (OOV) terms from clinical notes.

Model	IV F1	OOV F1
BERT	71.3	57.9
BERT-ADA	70.2	57.6
BERT-LIW	72.0	61.4
BERT-DAFT	74.9	63.6
BERT-DAFT-SYN	65.5	55.5

Table 3.7: Model performance on in-vocabulary (IV) and out-of-vocabulary (OOV) terms from doctor-patient conversations.

models identify correctly, we see that a large proportion (54.8%) contain one of three morphological patterns: (i) past tense verbs ending in "-ed", (ii) gerunds ending in "-ing", or (iii) nouns ending in "-tion" or "-sion". These patterns are also common among events in the news domain. For example, past tense verbs often refer to events that have already occurred and gerunds and nouns ending in "-tion" refer to processes. We hypothesize that BERT-based models might be exploiting these morphological regularities to correctly label unseen medical terms (e.g., irrigated, excision, dissected, wheezing, etc.). These patterns are more prevalent in notes (35.6%) than conversations (23.5%), explaining the surprising performance difference.

Performance on Various Event Types

We perform an additional type analysis with more fine-grained event types. For this analysis, we use the same typology as TimeML, with two additional labels for the event types we introduce. Following is the full set of event type labels:

1. **Occurrence:** Occurrence refers to all events describing something that happens or occurs in the world. This is the broadest class of events. For example, "I *took* Midol yesterday."

2. **Aspectual:** Aspectual events refer to events which focus on various aspects of a different event's history, such as initiation, termination, continuation etc. For example, "I *started* taking this medicine last Friday." Here *started* is an aspectual event describing the initiation of the event *taking*.

3. **Reporting:** Reporting events describe the action of an entity (person/group/organization) declaring something, narrating an event, providing information about an event etc. For example, "So you *said* you have been experiencing symptoms since yesterday?"

4. **Perception:** Perception events refer to events involving the physical perception of a different event. For example, "I *watched* my weight gain throughout the pregnancy."

5. **State:** States describe circumstances in which something obtains or holds true. For example, "My blood pressure is *higher* today". Note that annotation of state events in TimeML is subject to certain rules as outlined in §3.3.2.

6. **Intensional Action (I-Action):** Intensional actions introduce an explicit event argument describing an action or situation, from which we can infer something given its relation with the intensional action. For example, "We will *investigate* your symptoms further via this test." Here *investigate* is an intensional action associated with the *symptoms* event.

7. **Intensional State (I-State):** Intensional states contain stative events that refer to alternative or possible worlds. For example, "You might observe *higher* blood pressure for a few days when you start taking this medicine."

8. **Activity Pattern:** Activity patterns, as explained in §3.3.2 refer to events that are not clearly grounded to a single occurrence in time, but are still considered events since the presence of a participant stops them from being purely generic. For example, "You should *take* your blood pressure regularly."

9. **Long Term State:** Long-term states expand the annotation of states beyond TimeML restrictions, allowing the inclusion of long-term chronic conditions. For example, "You have a history of *COPD*."

We focus on fine-grained analysis of OOV tokens in particular, to study whether event types influence model performance on lexically shifted samples. We randomly sample ∼500 OOV tokens from each domain and label them for fine-grained event type. We run an ANOVA model with each token per model as an instance (total 5080 instances), noting Event Type, Target (notes/convos), Model (BERT/ADA/LIW/DAFT/DAFT-SYN) and Correctness (1 vs 0). Correctness is the dependent variable, while all others are independent variables. We include all pairwise interaction terms and the three way interaction between Event Type, Target and Model. We see a positive main effect of Event Type on Correctness ($F(6, 5010) = 332.5$, $p < .0001$) indicating that some event types are more difficult. There are two significant two-way interactions, one between Target and Event type ($F(6, 5010) = 7.72$, $p < .0001$), indicating that difficulty of event types differs across various target domains, and between Model and Event type ($F(24, 5010) = 2.12$, $p < .0001$), indicating that which model is better depends on event type. Three way interaction between Model, Event

type, and Target is also significant ($F(24, 5010) = 2.92$, $p < .0001$), indicating that performance differences between models per event type differs across various target domains.

Because the three-way interaction is significant, we interpret differences in performance per event type separately for each target domain using a student-t post-hoc analysis to determine which pairwise contrasts are statistically significant within this ANOVA model. This reveals that in clinical notes, our proposed method LIW outperforms all models on I-State events (i.e., hypothetical, future or negated states) and LongTermState events, a category never seen in the training data. These improvements might stem from the training algorithm used by LIW. LIW up-weights instances in news that resemble clinical data, which contains a high proportion of these event categories. Therefore, despite being infrequent in news, they get up-weighted, helping LIW identify them better.

3.4.5 Summary of Observations

- Unsupervised adaptation techniques help in building adaptable event extractors, especially for resource-scarce domains in the macro long tail. Our best-performing models attained F1 scores of 70.0 and 72.9 on clinical notes and conversations respectively, using *no* labeled target data. These models define a good low-bias starting point and can be further improved using few-shot learning.
- In accordance with our observations from the previous chapter, loss augmentation methods achieve the best performance on event extraction from clinical narratives. However, on clinical conversations, where syntactic and semantic shifts are more prominent than lexical, the pretraining method outperforms all other methods.
- Our proposed method LIW improves performance over a zero-shot baseline, but is not the best-performing adaptation method on both domains. However, this is still a step forward since our instance weighting method is capable of achieving positive transfer results, and even outperforms all other methods on LongTermState events, a category never seen in the training data.

3.5 Case Study II: Evaluating Domain-Aware Query Sampling for Active Learning

In this case study, our goal is to improve the performance of event extractors in a limited labeled data setting that allows models to use small amounts of labeled data from the target domain of interest. However, unlike the limited labeled data setting used in the case study in Chapter 2, we do not sample target instances at random. Instead, we explore the possibility of improving sample efficiency (i.e. achieving better performance using fewer target instances) by using active learning techniques to select target instances to obtain labels for. Active learning methods usually sample instances from the target domain that current models are most "uncertain" about, under

the assumption that training on such confusing instances is a sample-efficient strategy for model improvement.

In addition to exploring strong existing active learning baselines, we also propose a *domain-aware* query sampling strategy (DAQ), which incorporates distance from source domain instances into the sampling criterion while choosing target instances. The key idea behind this is that in a domain adaptation setting, sample efficiency of active learning can be further improved by ensuring that the process of sampling target instances avoids selecting ones that are highly similar to the source domain, since we can expect a model trained on the source domain to do reasonably well on them already. For DAQ, we experiment with two formulations of source-target similarity: (i) cosine similarity of a target instance to all source instances in a joint embedding space, and (ii) probability odds ratio of an instance belonging to the target domain according to a domain classifier. We run simulation experiments to evaluate the effect of incorporating these criteria during target instance selection, and whether this improves sample efficiency on two high-expertise domains that fall into the macro long tail: (i) clinical notes, and (ii) literary texts. Additionally, we also run simulation experiments on the task of named entity recognition from clinical narratives to develop a better understanding of the conditions under which various active learning methods improve sample efficiency.

3.5.1 Active Learning Baseline Sampling Strategies

Uncertainty Sampling with Representativeness (UNS)

Uncertainty sampling is one of the simplest and most frequently used sampling strategies for active learning (Lewis and Gale, 1994). This strategy works by selecting those instances that the current model is most uncertain about labeling. To quantify a model's uncertainty, prior work has developed various measures depending upon the nature of the task (e.g., classification, sequence labeling, regression, etc.). Some popularly used measures include least confidence sampling (Culotta and McCallum, 2005), margin sampling (Scheffer et al., 2001), and entropy (Shannon, 1948). For a comprehensive comparison of sampling strategies on several sequence labeling tasks, we refer interested readers to Settles and Craven (2008). Since we are treating event extraction as a token-level classification task, we choose token entropy as the measure to quantify uncertainty.[6] Given a target domain instance, consisting of a sequence of tokens $X_T = x_1, ..., x_N$, and a label space $m = 1, ..., M$, token entropy is defined as follows:

$$TE(X_T) = -\frac{1}{N}\sum_{n=1}^{N}\sum_{m=1}^{M} P(y_n = m)\log P(y_n = m) \tag{3.2}$$

[6]Preliminary experiments showed that the token entropy measure was on par with, and often performed better than least confidence sampling and total token entropy.

Note that $y_1, ..., y_N$ are the label predictions for each token in the sequence. The label probabilities $P(y_n = m)$ are typically obtained from the final softmax layer of the task model. A known caveat of uncertainty sampling methods is their tendency to choose outlier instances since they do not account for the underlying natural density of the data distribution (Settles, 2009). To handle this issue, we incorporate a *representativeness* criterion alongside token entropy. We use the criterion defined by Liao and Grishman (2011) which showed promising results on the event extraction task. Given a specific instance X_T from a set of target domain instances TD, its representativeness is measured as follows:

$$Repr(X_T) = \frac{1}{|TD| - 1} \sum_{X_k \in TD - X_T} sim(BERT(X_T), BERT(X_k)) \tag{3.3}$$

The $BERT(.)$ function in the above equation generates an embedding representation for an instance by running it through a pretrained language model and extracting the representation for the [CLS] token. $sim(.)$ is a function used to compute pairwise similarities between embedding representations of target domain instances; in all our experiments we use the cosine function. The uncertainty and representativeness scores for each target instance X_T are then combined as follows (as per Liao and Grishman (2011)):

$$UNS(X_T) = \lambda TE(X_T) + (1 - \lambda) Repr(X_T) \tag{3.4}$$

We use the same setting for λ as prior work.

Query-By-Committee (QBC)

Another simple and frequently used sampling strategy for active learning is the query-by-committee strategy (Seung et al., 1992). This strategy works by maintaining a committee of several task models $C = \{M_1, ..., M_C\}$, all of which are trained on the set of target instances chosen for labeling so far. The models in the committee typically represent competing hypotheses and the instances that they disagree the most on are considered most informative and sampled for annotation during the next iteration. Prior work has proposed several methods for choosing models for the committee such as random sampling from a posterior distribution (for generative models) (Dagan and Engelson, 1995; McCallum and Nigam, 1998), employing ensemble learning methods like bagging and boosting (Abe and Mamitsuka, 1998), encouraging diversity within ensembling (Melville and Mooney, 2004), and using cross-view training (Chaudhary et al., 2021). However, there isn't a clear consensus on which method results in the best choice of models for the committee, or what committee size is most appropriate.

In our work, we choose a committee of three models, each using a language model pretrained differently on unlabeled data from source and target domains, for representation computation. The first language model is an off-the-shelf pretrained model, the second language model is an

off-the-shelf model that is pretrained further on the target domain (similar to Chapter 2), and the third language model is an off-the-shelf model that is pretrained further on a mixture of data from source and target domains (similar to DAFT). Due to varying pretraining strategies, these models have differing "views" of target domain instances, and can represent competing hypotheses. We refer to these models as M_{base}, M_{target}, and M_{mix} throughout this chapter.

In addition to committee-building methods, prior work has also focused on developing strategies to measure level of disagreement between the models. Commonly used disagreement measures include average Kullback-Leibler divergence between the predictions of a single model and the "consensus" probability across all models (McCallum and Nigam, 1998), and vote entropy (Dagan and Engelson, 1995). Settles and Craven (2008) observe that vote entropy-based methods (especially sequence-level variants) tend to perform better than KL-divergence for several sequence labeling tasks. Therefore, we use a probabilistic vote entropy to measure disagreement. Given a target domain instance consisting of a sequence of tokens $X_T = x_1, ..., x_N$, a label space $m = 1, ..., M$ and a committee of 3 models $C = \{M_{base}, M_{target}, M_{mix}\}$, probabilistic vote entropy is computed as follows:

$$QBC(X_T) = -\frac{1}{N}\sum_{n=1}^{N}\sum_{m=1}^{M}\frac{P(y_n = m)}{|C|}\log\frac{P(y_n = m)}{|C|} \tag{3.5}$$

Again, $y_1, ..., y_N$ are the label predictions per token. Additionally, $P(y_n = m) = P_{M_{base}}(y_n = m) + P_{M_{target}}(y_n = m) + P_{M_{mix}}(y_n = m)$, which is the sum of the label probabilities produced by all models from the committee.

3.5.2 Incorporating Domain-Awareness Criteria

As described earlier, in addition to strong active learning baselines, we propose a *domain-aware* sampling strategy (DAQ), which incorporates distance from source domain instances while choosing target instances to label, in order to improve sample efficiency in a domain adaptation setting. We compute distance from source domain instances using two different formulations.

Classifier Confidence Formulation (DAQ-CC)

The first formulation we experiment with leverages confidence scores from a classifier model trained on the task of identifying whether instances belong to the source or target domains. The idea of using a domain discriminator to improve sampling when active learning is used in a domain adaptation setting has been explored in prior work. This idea was first tested by Rai et al. (2010), who used the domain discriminator to filter out target domain instances that are highly similar to the source domain. Their approach achieved promising improvements on sentiment analysis. Follow-up work experimented with alternate formulations such as using probability scores from a domain discriminator to re-weight target domain instances, but primarily evaluated their efficacy on computer vision tasks (Su et al., 2020; Fu et al., 2021). Motivated by this work, we adopt a

formulation that uses the probability odds ratio of an instance belonging to the target domain as measured by the domain discriminator, to compute weights for target domain instances. Given a target instance X_T and a domain discriminator D, the probability odds ratio is computed as follows:

$$OR(X_T) = \frac{P_D(y = t|X_T)}{1 - P_D(y = t|X_T)} \tag{3.6}$$

Note that $P_D(y = t|X_T)$ indicates the probabilities of instance X_T belonging to the target domain t according to the domain discriminator. This formulation results in higher weights for instances that are more likely to belong to the target domain as per the discriminator, which we hope improves sample efficiency by pushing active learning methods to avoid selection of instances too similar to the source domain. This weighting criterion is incorporated into baseline active learning sampling strategies (UNS and QBC) as follows:

$$DAQ - CC(X_T) = OR_(X_T) * BaseCriterion(X_T) \tag{3.7}$$

where $BaseCriterion(X_T)$ can be chosen to be $UNS(X_T)$ or $QBC(X_T)$.

Cosine Similarity Formulation (DAQ-CS)

The second distance formulation we experiment with leverages similarity between source and target domain instances in an embedding space. The motivation behind this formulation comes from prior work on using unsupervised clustering in embedding spaces to identify domains. Aharoni and Goldberg (2020) demonstrate that using pretrained language models to compute embedding representations of instances from different domains, followed by unsupervised clustering, successfully groups instances according to their domains. This indicates that similarity in a language model embedding space might serve as a strong alternative to the classifier confidence-based formulation for the task of re-weighting target domain instances. Therefore, we embed all source and target instances using a pretrained language model and then compute the average similarity of a specific target instance to all source instances. Given a target instance X_T, a set of source domain distances SD, and an language model embedding function $BERT(.)$, the average similarity score is computed as follows:

$$AS(X_T) = \frac{1}{|SD|} \sum_{X_k \in SD} sim(BERT(X_T), BERT(X_k)) \tag{3.8}$$

The $sim(.)$ function in the above equation is set to use cosine similarity in all our experiments. This equation produces higher scores for target domain instances that are more similar to source domain data, hence this weighting criterion is incorporated into baseline active learning sample strategies as follows:

$$DAQ - SC(X_T) = \frac{BaseCriterion(X_T)}{AS(X_T)} \tag{3.9}$$

where $BaseCriterion(X_T)$ can be chosen to be $UNS(X_T)$ or $QBC(X_T)$.

3.5.3 Experimental Setup

Datasets

For our simulation experiments on the event extraction task, we use the following datasets:[7]

- **Clinical Notes:** For the clinical notes domain, we use the i2b2 2012 dataset consisting of discharge summaries annotated with events (Sun et al., 2013). Unfortunately, we cannot use our MTSamples dataset since it does not have any associated training data.
- **Literary Texts:** For the literary text domain, we use the LitBank dataset (Sims et al., 2019), consisting of literary texts from Project Gutenberg annotated with events.

For our simulation experiments on named entity recognition from clinical narratives, we use the same set of datasets as the case study from the previous chapter: i2b2 2006 (Uzuner et al., 2007), i2b2 2010 (Uzuner et al., 2011), and i2b2 2014 (Stubbs and Uzuner, 2015). As described earlier, all datasets consist of discharge summaries. The i2b2 2006 and i2b2 2014 datasets focus on the de-identification task, and are annotated with PHI (private health information) entities such as patient names, doctor names, hospitals, etc. The i2b2 2010 dataset is annotated with medical entities of three types: problems, tests, and treatments. Note that for all NER datasets, models are evaluated in a coarse setting, in which the model is only expected to detect entities, without any entity type prediction. Appendix C presents some examples of annotated instances from all these datasets.

Baseline Task Model

For our baseline task model, we choose a strong BERT-based sequence labeling model. This model computes token-level representations using a BERT encoder followed by a linear layer that predicts labels for every token. For all active learning experiments, this model is first trained on the source dataset, and label probabilities from this source-trained model are then used for various active learning strategies. For every iteration of active learning, the source-trained model is further finetuned on all target instances chosen until that point. This is slightly analogous to the SC->TG baseline in the case study from Chapter 2, in that we are continuously performing target-specific finetuning of a source-trained model. Note that it is possible to use other supervised adaptation techniques such as multi-task training, frustratingly easy domain adaptation, etc., instead of finetuning on target data. However, our results from the previous chapter demonstrated that SC->TG often achieves comparable performance to supervised adaptation methods, so we use this technique to avoid introducing an additional dimension of method combination.

[7] We omit the domain of clinical conversations from these experiments since we no longer have access to the proprietary dataset from Abridge.

Method	LitBank			i2b22012		
	P	R	F1	P	R	F1
TG	74.70	60.56	66.89	87.97	88.31	88.14
SC->TG	68.60	68.37	68.48	83.63	**92.15**	87.68
SC+TG	64.45	60.37	62.34	84.31	91.76	87.88
Rand	**77.09**	61.35	68.32	87.20	89.52	88.34
UNS	69.94	71.63	70.77	88.28	87.53	87.90
+DAQ-CC	72.12	69.67	70.88	85.30	89.23	87.22
+DAQ-CS	74.05	68.88	71.37	**89.28**	88.03	**88.65**
QBC	68.33	**75.77**	71.86	88.32	86.46	87.38
+DAQ-CC	69.00	73.40	71.13	86.73	87.63	87.18
+DAQ-CS	68.78	75.30	**71.89**	88.03	88.74	88.38

Table 3.8: Final performance of all models on event extraction datasets. Note that for active learning variants, we report the performance after 20 iterations of active learning. The TG, SC->TG and SC+TG baselines are described in detail in Section 3.5.3, while Rand refers to a baseline which randomly samples additional instances at each iteration instead of choosing them via active learning. UNS and QBC refer to uncertainty sampling and query-by-committee strategies respectively, which DAQ-CC and DAQ-CS refer to the classifier confidence and cosine similarity formulations of our domain-awareness criteria.

In addition to various active learning variants, we also report the performance of the same set of limited data baseline methods that we evaluate in the previous chapter: TG (training on target domain data only), SC->TG (training source data, followed by target data), and SC+TG (joint training on both source and target data). Note that all these baselines are trained passively, i.e., we randomly sample a set of target domain instances for training.

Hyperparameter Details

We use the cased version of BERT-base for all our experiments. For all passively trained limited data baselines, we follow the same setting as the previous chapter and randomly sample 1000 target domain instances. For all active learning variants, we sample batches of 50 target instances at every iteration, and continue learning for 20 iterations. At each AL iteration, the model is further finetuned on the current set of labeled target data for 2 epochs, with a learning rate of 2e-5. Before starting the active learning process, the model is trained on source domain data for 20 epochs, with a learning rate of 2e-5 and early stopping.

Method	i2b22006			i2b22010			i2b22014		
	P	R	F1	P	R	F1	P	R	F1
TG	79.79	84.78	82.21	76.33	76.67	76.50	84.87	84.63	84.75
SC->TG	86.82	90.39	88.57	71.01	74.42	72.67	89.12	79.25	83.88
SC+TG	80.08	74.89	77.40	78.18	70.92	74.38	79.39	64.99	71.47
Rand	**94.06**	**93.00**	**93.52**	79.56	77.79	78.67	89.18	86.37	87.75
UNS	89.49	77.60	83.12	60.96	65.82	63.30	85.72	93.16	89.29
+DAQ-CC	65.54	82.18	72.92	70.02	79.96	74.66	**92.26**	91.21	**91.73**
+DAQ-CS	91.81	87.81	89.77	**81.37**	80.84	**81.11**	91.20	91.90	91.55
QBC	75.19	83.38	79.08	64.16	73.80	68.64	72.94	93.23	81.84
+DAQ-CC	69.35	70.33	69.84	77.27	75.32	76.28	91.29	91.28	91.29
+DAQ-CS	89.63	91.67	90.64	77.46	**82.10**	79.71	88.65	**93.87**	91.19

Table 3.9: Final performance of all models on named entity recognition datasets, in the coarse setting. Note that for active learning variants, we report the performance after 20 iterations of active learning. The TG, SC->TG and SC+TG baselines are described in detail in Section 3.5.3, while Rand refers to a baseline which randomly samples additional instances at each iteration instead of choosing them via active learning. UNS and QBC refer to uncertainty sampling and query-by-committee strategies respectively, which DAQ-CC and DAQ-CS refer to the classifier confidence and cosine similarity formulations of our domain-awareness criteria.

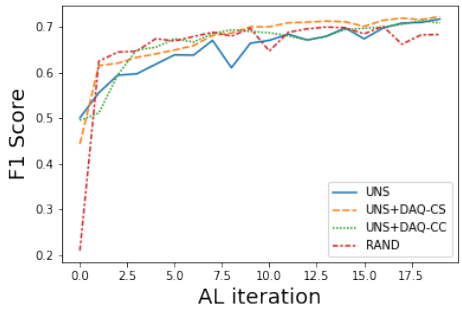
3.5.4 Results

Final Performance

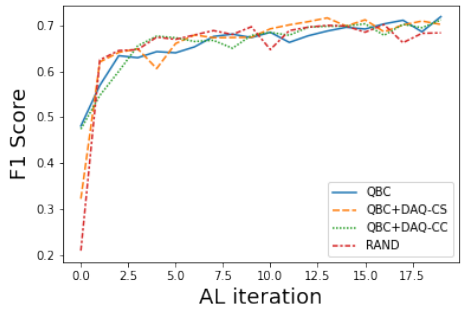
Tables 3.8 and 3.9 show the performance of all passive baselines and active learning variants on event extraction and NER datasets respectively. Note that for active learning variants, we report the final performance after 20 iterations of active learning have been completed. From these tables, we make the following major observations:

- Adding the domain-awareness criterion, specifically the cosine similarity formulation (DAQ-CS), helps to further improve performance of active learning baselines. On the event extraction datasets, these gains are modest ($\sim$1 F1 point). However, the gains are much more pronounced on the NER datasets ($\sim$4-18 F1 points).

- The random sampling baseline achieves strong performance, and even outperforms all active learning variants on the i2b2 2006 NER dataset, which indicates the utility of choosing a strong starter model trained on source domain data. It is also interesting to note that random sampling beats the TG baseline, demonstrating its data efficiency (since both use the same number of instances).

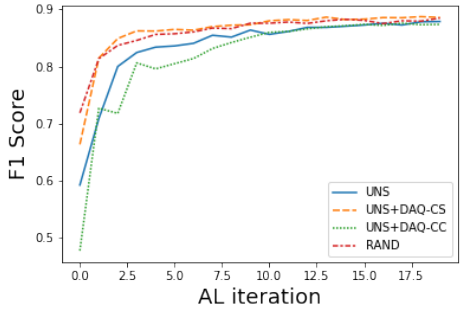


(a) Performance of all uncertainty sampling variants.

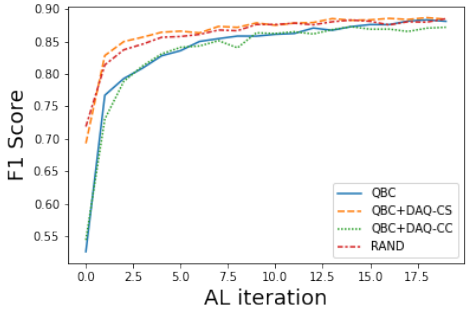


(b) Performance of all query-by-committee variants.

Figure 3.5: Per-iteration performance of various active learning methods, and the random sampling baseline, on event extraction from the LitBank dataset.



(a) Performance of all uncertainty sampling variants.



(b) Performance of all query-by-committee variants.

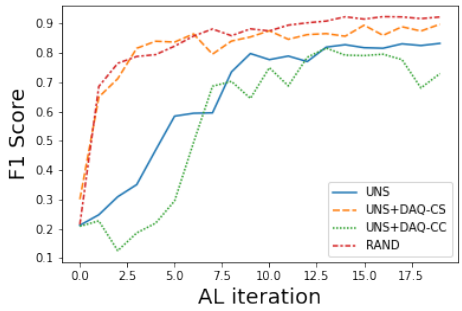
Figure 3.6: Per-iteration performance of various active learning methods, and the random sampling baseline, on event extraction from the i2b2 2012 dataset.

In addition to analyzing performance at the end of 20 active learning iterations, we also look at model performance at every iteration to gain a better understanding of its evolution.

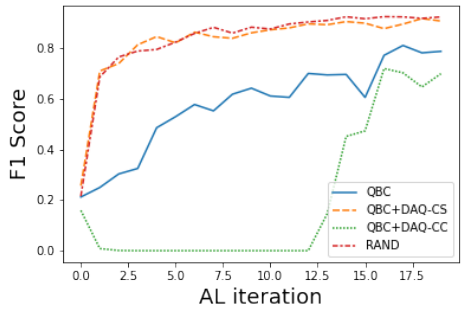
Simulation Graphs

Figures 3.5a and 3.5b show the per-iteration performance of all uncertainty sampling variants and query-by-committee variants on the LitBank dataset. Similarly, Figures 3.6a and 3.6b show per-iteration performance on i2b2 2012, Figures 3.7a and 3.7b show per-iteration performance on i2b2 2006, Figures 3.8a and 3.8b show per-iteration performance on i2b2 2010, and Figures 3.9a and 3.9b show per-iteration performance on i2b2 2014. Note that we also include the random sampling baseline performance in all graphs.

From these graphs, we continue to see that our similarity-based formulation of domain awareness helps on most datasets, and starts improving the performance of active learning baselines early on (with the exception of i2b2 2014 and Litbank where results are mixed). These gains appear to

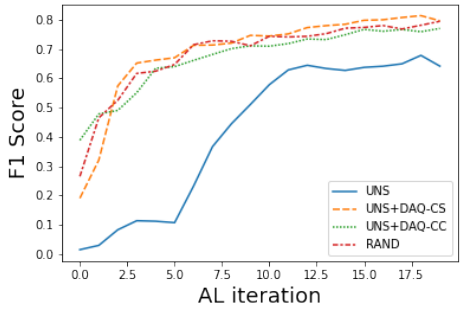


(a) Performance of all uncertainty sampling variants.

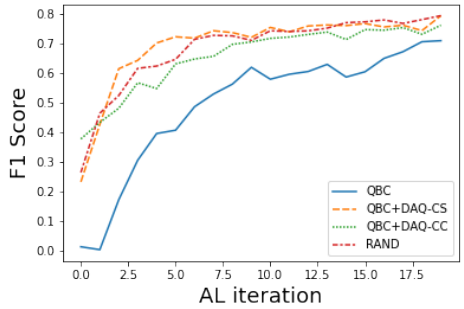


(b) Performance of all query-by-committee variants.

Figure 3.7: Per-iteration performance of various active learning methods, and the random sampling baseline, on entity extraction from the i2b2 2006 dataset.



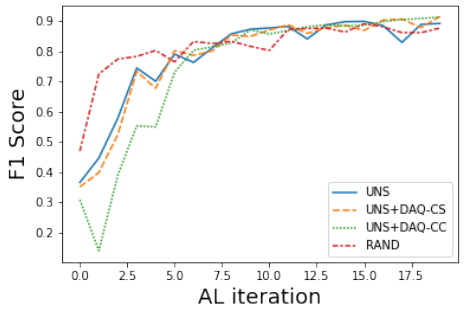
(a) Performance of all uncertainty sampling variants.



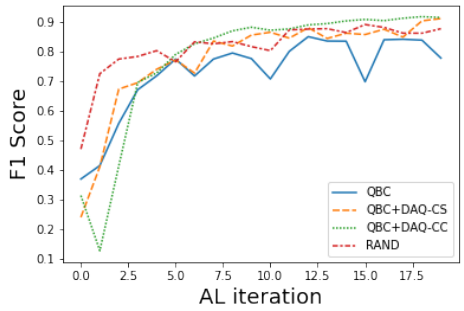
(b) Performance of all query-by-committee variants.

Figure 3.8: Per-iteration performance of various active learning methods, and the random sampling baseline, on entity extraction from the i2b2 2010 dataset.

be particularly pronounced on the i2b2 2006 and i2b2 2010 datasets. Therefore, DAQ-CS seems to be a strong contender when dealing with extremely low annotation budgets (e.g., <200 instances). In subsequent analyses, we perform a deeper investigation into how different properties of datasets such as label sparsity and source-target distance influence the utility of the domain-awareness criteria. Lastly, as with final iteration performance, we note that random sampling is an extremely powerful baseline, achieving similar performance as the best-performing active learning variant on every dataset. This seems to suggest that our source-trained BERT-based starter model is already data-efficient and does not benefit much from smarter target instance sampling.

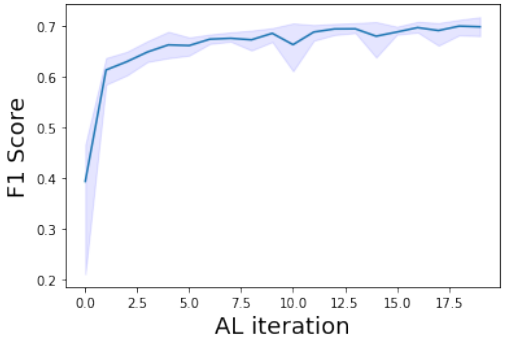


(a) Performance of all uncertainty sampling variants.

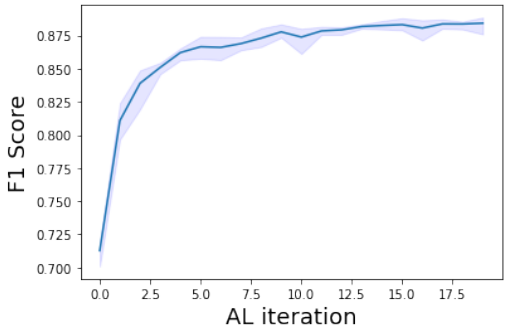


(b) Performance of all query-by-committee variants.

Figure 3.9: Per-iteration performance of various active learning methods, and the random sampling baseline, on entity extraction from the i2b2 2014 dataset.



(a) Performance on LitBank.



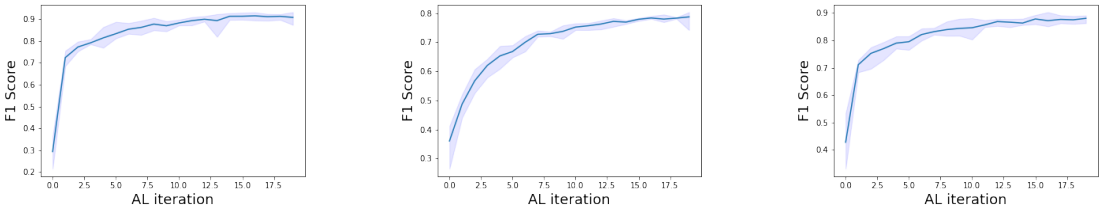
(b) Performance on i2b22012.

Figure 3.10: Variation in performance of random sampling baseline on various event extraction datasets upon using different seeds for initialization. The line graph indicates average performance at each active learning iteration, while the shaded region indicates minimum and maximum performance observed across runs.

3.5.5 Analysis and Discussion

Variation in Random Sampling Baseline Performance

The strong performance of our random sampling baseline raises a natural question: is the performance of this baseline consistent across varying seed values? This is especially pertinent for our setting because prior work has shown that finetuning pretrained language models, particularly on small datasets, is highly brittle and can lead to substantially different results on varying the random seed value (Dodge et al., 2020). To study variation in random sampling baseline performance, we re-run this baseline with five different seed values on all datasets. Figures 3.10a and 3.10b show the average, minimum and maximum performance from these runs on both event extraction datasets. Similarly, Figures 3.11a, 3.11b, and 3.11c show the average, minimum and maximum performance from these runs on all NER datasets. From these graphs, we can see that during early



(a) Performance on i2b22006. (b) Performance on i2b22010. (c) Performance on i2b22014.

Figure 3.11: Variation in performance of random sampling baseline on various NER datasets upon using different seeds for initialization. The line graph indicates average performance at each active learning iteration, while the shaded region indicates minimum and maximum performance observed across runs.

iterations, performance variation is often massive. For example, the difference between minimum and maximum performance on i2b22010, i2b22014, and LitBank is >=20 F1 points during early iterations. However this variation stabilizes over time, indicating that final performance scores for the random sampling baseline are quite consistent across seed values.

Variation in Performance across Entity Types

Similar to the previous chapter, in addition to looking at overall performance, we look at the performance of all NER models on various entity types. However, as noted earlier, all models in these experiments are only trained in a coarse setting, i.e. they only identify entity spans and are not trained to predict entity types. Given this constraint, we approximate performance on an entity type (e.g., location) by computing recall as follows:

$$R_{location} = \frac{\text{\#Number of location entities from the gold standard identified correctly by the model}}{\text{\#Total number of location entities in the gold standard}} \tag{3.10}$$

Note that *correct identification* here only refers to the model identifying span boundaries for the entity correctly.

Table 3.10 presents the recall scores per entity type for all active learning variants and baselines, as computed by this metric. From this table and Table 3.9, we can immediately see that on i2b22006 and i2b22014, the model that achieves highest recall on the MISC category, which is the most populous one, also achieves best overall recall scores. However, these models are not necessarily the best-performing ones on all other entity types. In particular, we note that domain-aware model variants often achieve strong performance on entity categories that differ significantly from the source dataset (CoNLL-2003), but are not as populous in the target dataset. For example, in i2b22006, as discussed in the previous chapter, location entities often comprise of complete addresses (not common in CoNLL-2003) and only comprise $\sim$12% of the data. This may partly be due to the selectivity of the domain-aware sampling strategies, which tend to emphasize that samples should simultaneously be distant from the source domain and difficult for the current

Method	i2b22006				i2b22010			i2b22014		
	PER	LOC	ORG	MISC	PROB	TEST	TREAT	PER	LOC	MISC
Rand	87.61	60.50	**88.31**	**96.87**	76.07	77.89	78.99	74.92	55.10	59.98
UNS	66.77	50.42	64.65	85.46	69.70	57.40	68.04	**79.71**	74.19	62.33
+DAQ-CC	78.25	52.94	43.20	92.79	80.43	75.56	82.62	75.48	73.47	61.48
+DAQ-CS	87.68	75.63	52.66	95.31	79.40	**81.72**	80.85	79.29	74.35	60.63
QBC	**89.28**	60.50	**88.31**	80.16	75.19	69.26	75.41	79.47	**76.01**	62.36
+DAQ-CC	40.68	60.50	48.23	87.43	80.66	61.85	80.44	74.68	74.90	61.43
+DAQ-CS	87.68	**80.67**	78.85	95.92	**82.18**	79.87	**83.09**	78.56	73.75	**64.20**

Table 3.10: Recall scores per entity type for all active learning variants on named entity recognition datasets. Note that these scores are recorded after 20 iterations of active learning. Rand refers to the baseline which randomly samples additional instances at each iteration instead of choosing them via active learning. UNS and QBC refer to uncertainty sampling and query-by-committee strategies respectively, which DAQ-CC and DAQ-CS refer to the classifier confidence and cosine similarity formulations of our domain-awareness criteria.

model. We also see that domain-aware variants are the best-performing models on all entity types in i2b22010, which contains entity types that are never found in the source dataset. These observations indicate that domain-awareness seems to provide some utility, and we explore this further in subsequent analyses.

Effect of Label Sparsity on the Utility of Domain-Awareness

Based on final performance and per-iteration performance graphs, we can see that the domain-awareness criterion is typically helpful, except for i2b2 2014 and Litbank. To better identify properties of various datasets that make the domain-awareness criterion more/less useful, we first look at label sparsity across all datasets. Table 3.11 shows the percentage of tokens that are labeled as entities or events for all datasets used in our experiments. From this table, we can see that i2b2 2014 and LitBank are the most label-sparse datasets. Strategies such as uncertainty sampling and query-by-committee are known to work better under sparsity, since under high-sparsity conditions, these strategies tend to be most uncertain about fewer spans, which typically correspond to events/entities. Hence, these strategies are already strong baselines, and may not benefit much from the addition of domain-awareness. Interestingly, the i2b2 2006 dataset is also quite sparse, but still benefits from domain-awareness, indicating that label sparsity is only a partial determinant of model performance.

Effect of Source-Target Distance on the Utility of Domain-Awareness

In addition to label sparsity, we investigate whether distance between source and target datasets influences how well the domain-awareness criterion works. For this analysis, we use the same general family of divergence measures (information-theoretic measures and term vocabulary

Dataset	Task	Label Proportion
TimeBank	EE	11.31
LitBank	EE	3.73
i2b2 2012	EE	35.47
CoNLL 2003	NER	16.84
i2b2 2006	NER	5.34
i2b2 2010	NER	23.89
i2b2 2014	NER	4.11

Table 3.11: Percentage of tokens labeled as entities/events across all datasets used in our experiments.

	CoNLL-2003			TimeBank	
	i2b22006	**i2b22010**	**i2b22014**	**i2b22012**	**LitBank**
TVO	0.1583	0.1335	0.1485	0.1536	0.1794
KLD	1.3145	1.1664	1.0658	1.0997	0.8505
JSD	0.2468	0.2309	0.2134	0.2437	0.1975
RD	1.2998	1.1539	1.0547	1.0917	0.8441

Table 3.12: Distance between source-target domain pairs used in our case study according to various label-aware measures. As indicated in the table, for i2b22006, i2b22010 and i2b22014, distance is computed from CoNLL-2003, while for i2b22012 and LitBank, distance is computed from TimeBank. Note that for TVO, lower values mean higher source-target distance, while higher values correspond to higher source-target distance for all other measures.

overlap) as the case study from the previous chapter. However, we make a key tweak to all these measures: incorporating label information. The vanilla variants of these measures treat word types as random variables, computing distance as word type overlap or divergence between word type probability distributions, while ignoring changes in word type-label association across domains. To account for these changes as well, we treat word type-label pairs as random variables, resulting in label-aware variants of the same set of measures. The distance between various source-target domain pairs used in our experiments under these label-aware measures is shown in Table 3.12. Note that unlike vanilla variants, label-aware measures cannot be used to predict performance in advance since we do not typically have access to labels for the target domain. However, they can be used for retrospective analyses to better understand the behavior of various adaptation methods.

To compute correlation between domain-awareness and domain distance, we first compute percentage change in performance (improvement or drop) achieved by adding domain-awareness to an active learning baseline (UNS or QBC) across all datasets. We compute performance changes separately for both formulations explored in our experiments. After computing percentage changes for each formulation, we calculate the Pearson correlation between these values and the source-target distance according to each measure. Table 3.13 shows the results from this correlation

Formulation	TVO	KLD	JSD	RD
DAQ-CC	-0.7131	0.2991	0.1787	0.2964
DAQ-CS	-0.6993	0.1575	0.0704	0.1551

Table 3.13: Correlation between performance improvements/drops on adding domain-awareness (recorded as percentage change over UNS/QBC baseline scores) and label-aware source-target domain distance for each distance formulation. Note that performance changes are averaged over all 20 active learning iterations.

analysis. From this table, we can see that most information-theoretic measures do not show strong correlation with performance changes, but term vocabulary overlap (TVO) shows strong negative correlation with domain-awareness. This indicates that domain-awareness is likely to provide higher benefits on source-target domain pairs with low overlap (i.e., domain pairs with fewer overlapping word type-label pairs). In addition to improving our understanding of dataset properties that influence the utility of domain-awareness, this provides some evidence that our domain-awareness formulation helps bridge larger label drifts through its focus on choosing target instances that are distant from the source domain.

The Limits of Active Learning

Despite establishing the utility of our domain-awareness criterion, we observe that none of the active learning methods are able to outperform a random sampling baseline, both on final performance as well as per-iteration performance. This appears to be in contrast to much prior work that has established the utility of active learning for low-resource settings (Ambati, 2011; Chaudhary et al., 2019, 2021), including sequence labeling tasks for high-expertise domains like clinical text (Chen et al., 2015; Shelmanov et al., 2019; Lybarger et al., 2021; Liu et al., 2022). However, there have been some instances in which active learning methods have shown inconsistent or no performance improvements. Lowell et al. (2019) conducted a study evaluating the utility of active learning methods on four text classification benchmarks, and three model families: SVMs, LSTMs and CNNs. Their experiments reveal two key observations. First, while active learning shows improvements on certain domains/tasks, overall benefits are inconsistent and no specific strategy is a clear winner, which is problematic in a real world setting in which practitioners may need to commit to one ahead of time. This observation has been echoed by other studies such as Settles and Craven (2008), who evaluated several strategies for sequence labeling. Second, when new models (also called successor models) are trained on data collected via active learning, they do not consistently outperform variants trained on data sampled in IID fashion. Moreover, despite better performance in simulation experiments, which is the most commonly used evaluation setting, the benefits of active learning often do not translate to cost savings during actual user studies (Settles et al., 2008; Chen et al., 2017c). A key difference to note is that these studies have focused on the typical active learning setting in which a model is being created for a new task/domain from scratch. However, our experiments evaluated the utility of active learning in an adaptation

setting, in which we have already warm-started our model by training on related data from other domains. Our observation that active learning methods do not offer much additional benefit over pretrained language models in this setting helps in revealing additional limitations of active learning techniques not explored by prior work.

3.5.6 Summary of Observations

- Adding a domain-awareness criterion typically helps boost the performance of strong active learning baselines in an adaptation setting. Of the two formulations we experiment with, our new embedding similarity-based formulation seems to achieve stronger performance across both event extraction and named entity recognition.
- Designing label-aware variants of source-target divergence measures and computing correlation with performance change on incorporating domain-awareness shows that this criterion is capable of bridging large drifts in word type-label overlap. In accordance with our observations from chapter 2, TVO again seems to be the most predictive metric.
- Active learning variants in general are unable to outperform a simple random sampling baseline in an adaptation setting. This indicates that use of pretrained language models and starting the active learning process with a source-trained model are already highly data-efficient tactics, and also helps identify a potential new failure setting for active learning.

3.6 Conclusion

In this chapter, we attempted to further develop our understanding of macro-level adaptation by expanding the set of macro dimensions studied so far. More specifically, in addition to clinical narratives, we brought two new domains under our purview: (i) literary texts, and (ii) transcripts of doctor-patient conversations. We also proposed two new adaptation methods:

- Likelihood-based instance weighting (LIW) (Naik et al., 2021b)
- Active learning with domain-aware query sampling (DAQ)

LIW is an unsupervised adaptation method from the hybrid instance weighting category that uses target domain language model likelihood to compute weights for source domain instances. DAQ is a data-centric active learning method, that adds an additional domain-awareness criterion during the query sampling process. We experimented with two different formulations of domain distance, based on classifier confidence (DAQ-CC) and embedding similarity (DAQ-CS). To understand the strengths and weaknesses of these newly proposed methods, as well as to further our understanding of existing adaptation methods, we conducted systematic case studies on event extraction datasets. For DAQ, we also conducted additional experiments on NER to better understand the effect of the domain-awareness criterion. Our experiments demonstrated promising results with both these adaptation methods. LIW improved performance over a zero-shot baseline on both the domains that

it was tested on, but did not outperform other unsupervised adaptation methods. DAQ improved performance over existing active learning baselines on most datasets, but no active learning variants were able to outperform a random sampling strategy. Based on extensive experimentation and supplementary quantitative and qualitative analyses, these case studies further extended our understanding of macro-level adaptation methods with the following observations:

- In an unsupervised adaptation setting, different method categories performed best depending on the linguistic nature of source-target domain shift. For event extraction, loss-centric methods seemed to be the best-performing category for high-expertise narrative domains (consistent with Chapter 2), while pretraining methods seemed to be the best-performing category for high-expertise non-narrative domains.
- Active learning methods did not improve performance or data-efficiency over a random sampling baseline in an adaptation setting, which could be a potential new failure case for this category of methods.
- The TVO measure of source-target domain divergence continued to be strongly correlated with performance improvements/drops achieved by adaptation methods.

CHAPTER 4

Improving Micro-Level Adaptation: A Case Study on Discourse-Level Event Ordering

In this chapter, we switch our focus to adaptation between micro-level long tail dimensions, i.e. adapting models to handle different linguistic phenomena under the same macro-dimensional scenario (same task, domain, language and adaptation setting). We delve into the problem of data-scarce micro-level adaptation for the task of temporal ordering of events, targeting event pairs that are far apart in text. Despite event ordering being an extensively studied problem, prior work has mostly focused on local pairs, i.e. ordering events present in the same or adjacent sentences, and sidelined distant event pairs, which might require models to learn to handle a different set of linguistic phenomena such as maintaining transitivity and chain reasoning. To address this gap, and simultaneously study micro-level adaptation, we make the following contributions:

- TDDiscourse, a new event ordering benchmark with a discourse-level focus (Naik et al., 2019)
- A joint BiLSTM+ILP model architecture that incorporates heuristics (e.g., transitivity) via loss augmentation for better adaptation (Breitfeller et al., 2021)

In order to ensure macro dimension consistency, our new discourse-level event ordering benchmark TDDiscourse is constructed by augmenting the existing TimeBank-Dense dataset (Cassidy et al., 2014), a corpus of English news articles, with more long-distance event pair annotations. Since sourcing expert annotation for all possible long-distance event pairs is expensive, we develop a heuristic algorithm for automatic inference of temporal relations for some pairs (TDD-Auto subset), and then obtain expert annotation for a subset of non-inferable pairs (TDD-

Man subset). Benchmarking multiple existing state-of-the-art models on TDDiscourse reveals its challenging nature. We then study the problem of unsupervised micro-level adaptation by training models on local event pairs (i.e. TimeBank-Dense) and evaluating them on distant event pairs (i.e. TDDiscourse). We evaluate adaptation methods from two categories: (i) a pseudo-labeling data-centric method that adds heuristically labeled data from TDD-Auto during model training, and (ii) a loss augmentation model-centric method that incorporates heuristics such as transitivity into the loss function via an integer linear programming (ILP) constraint framework. Our results show that both methods improve performance over a zero-shot baseline, but combining methods does not always lead to consistent performance boosts. Our observations from this case study highlight interesting future research avenues that can be explored to develop better techniques for micro-level adaptation.

4.1 Introduction

Temporal ordering of events is a crucial problem in automated text analysis. Systems capable of performing this task can find widespread applicability in downstream tasks such as time-aware summarization, temporal information extraction or event timeline construction. Prior work has focused extensively on creating annotated benchmark corpora for the task of temporal ordering, some notable efforts being the development of the TimeML annotation schema (Pustejovsky et al., 2003a), TimeBank (Pustejovsky et al., 2003b) and TimeBank-Dense (Cassidy et al., 2014). However, most benchmarks have focused mainly on local ordering, i.e., ordering events present in the same or adjacent sentences, which is fairly restrictive. As pointed out by Reimers et al. (2016), this allows systems which simply rely on explicit local syntactic cues to achieve moderate performance. On the other hand, global ordering, i.e. ordering events which are more than one sentence apart requires models to employ implicit reasoning such as maintaining discourse-level (global) consistency, understanding probable causal/prerequisite relationships and performing chain reasoning at the document-level. State-of-the-art temporal ordering models are rarely exposed to examples that require such reasoning due to lack of global (or long-distance) annotations in benchmark datasets. In other words, despite their utility and challenging nature, long-distance event pair examples have been relegated to the long tail of temporal ordering research. Therefore, in this chapter, we specifically focus on the task of ordering long-distance event pairs, while also using it as a testbed to study adaptation between micro-level long tail dimensions.

To encourage research on long-distance examples requiring implicit reasoning, we first construct TDDiscourse, a new benchmark dataset focused on discourse-level temporal ordering. In addition to a discourse-level focus, another requirement for this benchmark is to ensure its utility as a testbed for micro-level adaptation. To achieve this, we need to maintain consistency across the macro dimensions of task, domain, language and adaptation setting. We achieve this during the creation of TDDiscourse by augmenting TimeBank-Dense (Cassidy et al., 2014), an existing corpus of English news articles, with more long-distance event pair annotations. TimeBank-Dense and

TDDiscourse are thus consistent along all macro dimensions, but can contain different proportions of specific linguistic phenomena since the former focuses on local event pairs, while the latter focuses on long-distance pairs. Our work on constructing TDDiscourse makes the first attempt to *explicitly* annotate relations between event pairs that are more than one sentence apart, a more difficult annotation task than previous datasets. In addition to facing similar challenges as prior work (eg: hypothetical/negated events (Cassidy et al., 2014)), we tackle new *global discourse-level* issues such as incorporating event coreference and causality/prerequisite links arising from world knowledge during the annotation procedure. To handle these, we design a careful coding scheme that achieves high inter-annotator agreement (Cohen's Kappa of 0.69 on the test set). However, getting expert manual annotation for all possible long-distance event pairs is expensive. Moreover, it is possible to leverage annotations from existing datasets to automatically infer temporal relations for certain event pairs. To make optimal use of expert annotation, we develop a heuristic algorithm for automatic inference of temporal relations using EventTime (Reimers et al., 2016) and apply this to all documents. We validate our algorithm by obtaining human annotations for a subset of 100 examples and observing agreement with the generated label in 99% cases. We then randomly subsample the unannotated event pairs and source expert annotations for those. At 6150 pairs, our manually annotated subset (TDD-Man) is of the same size as TimeBank-Dense. Adding the automatically inferred subset (TDD-Auto) makes our dataset 7x larger (§4.3.3). Finally, we perform a principled comparison between event pairs from the manual and automatic subsets by annotating 3 test documents (107 manual and 110 automatic event pairs) with linguistic phenomena required to reason correctly about the pair. These annotations suggest that our manual subset in particular exhibits a high proportion of global discourse-level linguistic phenomena such as reasoning about chains of events.

In addition to developing TDDiscourse, we establish the challenging nature of this dataset by benchmarking the performance of several state-of-the-art temporal ordering models that achieve high performance on TimeBank-Dense. This includes a model that tries to improve performance on long-distance examples by enforcing discourse-level consistency via explicit transitivity rules framed as integer linear programming (ILP) constraints in a structured perceptron (SP). Most other SOTA models are non-transitive, making separate local ordering decisions for each event pair, which may result in global inconsistency. For example, for events A, B and C, if A occurs before B and B occurs before C, transitivity implies that A occurs before C. But models classifying each pair independently may assign a different relation to A-C. Incorporating transitivity rules as ILP constraints attempts to correct for this by biasing models to prefer predictions that are consistent at the discourse-level. The SP+ILP model was initially proposed by Ning et al. (2017), but we design a stricter ILP formulation in order to improve tractability on our data, which contains 7x more TLINKs. We also benchmark three other state-of-the-art models on TimeBank-Dense on our data, after introducing minor modifications for discourse-level temporal ordering, reporting scores on TDD-Auto and TDD-Man separately. We observe that models perform worse on average on TDDiscourse as compared to TimeBank-Dense, with none beating a majority class baseline

on TDD-Man. Notably, incorporating transitivity rules helps improve both overall performance as well as prediction consistency. A manual analysis of model errors on TDD-Man reveals key shortcomings of these SOTA temporal ordering techniques. These experiments indicate that our dataset[1] serves as a challenging new resource for the temporal ordering community, and insights from our analysis highlight key areas for future research in building more global discourse-aware models.

We then use TDDiscourse as a testbed to conduct a case study on unsupervised micro-level adaptation by training a model on TimeBank-Dense, which consists of local event pairs, and evaluating its performance on long-distance event pairs from TDDiscourse. The task model used for this case study is a dependency parse-based BiLSTM, one of the SOTA models on TimeBank-Dense. To improve micro-level adaptation, we develop a joint neural model (BiLSTM+ILP), which infuses pre-defined heuristics into the BiLSTM model via integer linear programming (ILP) constraints in a structured support vector machine (SSVM) framework. Since ILP constraint infusion is carried out by modifying the model's loss function, this proposed adaptation method falls under the model-centric coarse category and the loss augmentation fine category in our adaptation method taxonomy. Motivated by its observed utility from the previous benchmarking experiment, one of the key heuristics that we incorporate using ILP is transitivity. Additionally, we also evaluate the effect of adding constraints generated from explicit textual semantic cues by a rule-based temporal parser called STAGE (Semantic Temporal Alignment Grammatical Extraction), during the adaptation process. Aside from the BiLSTM+ILP model, we also evaluate the performance of a data-centric method from the pseudo-labeling fine category that adds heuristically labeled training data from TDD-Auto while training the task model. Note that since TDD-Auto is entirely automatically generated using a temporal inference algorithm, using TDD-Auto as additional training data during adaptation can be viewed as a form of distant supervision, which puts this method under the pseudo-labeling category. The complete experimental setup for this case study is summarized below:

- **Task:** Temporal ordering, or temporal relation classification (text classification)
- **Source Dataset:** Local event pairs (TimeBank-Dense) (Cassidy et al., 2014)
- **Target Dataset(s):** Long-distance event pairs (TDDiscourse) (Naik et al., 2019)
- **Task Model:** Dependency parse-based BiLSTM classifier (Cheng and Miyao, 2017)
- **Adaptation Method:** Incorporating heuristics (e.g., transitivity) via ILP constraints (model-centric method) (Breitfeller et al., 2021)
- **Adaptation Baseline(s):** Incorporating heuristically generated data from TDD-Auto (data-centric method)
- **Adaptation Setting:** Unsupervised

From this case study, we observe that both adaptation methods improve the performance of the baseline BiLSTM task model. Among the two methods compared, the pseudo-labeling method (i.e., incorporating TDD-Auto training data) provides larger performance gains. This can partly

[1]Dataset is available at: https://github.com/aakanksha19/TDDiscourse

be attributed to the fact that while ILP constraints are designed to handle specific categories of linguistic phenomena, adding TDD-Auto data provides the model access to instances exemplifying the entire range of linguistic phenomena present in long-distance pairs. We also experiment with combining both methods, and observe slightly higher gains than using one method on TDD-Man, but no improvements on TDD-Auto.

Interestingly, the pseudo-labeling method, trained on TimeBank-Dense+TDD-Auto data, performs worse on the TDD-Auto test set than a model trained only on TDD-Auto data, indicating the presence of conflicting instances in the TimeBank-Dense dataset. However, designing an unsupervised instance weighting/data selection technique to identify such instances is not straightforward in the micro-level adaptation setting. Unlike techniques such as likelihood-based instance weighting (LIW) and classifier-based instance weighting that were used in previous chapters, we cannot rely on largely lexical similarity to learn source-target similarity when macro dimensions are consistent. This presents an interesting question for future research on developing better methods for micro-level adaptation to tackle.

4.2 Background

4.2.1 Temporal Ordering Datasets

The development of TimeML (Pustejovsky et al., 2003a) and TimeBank (Pustejovsky et al., 2003b) marked the first attempt towards creating a corpus for temporal ordering of events. TimeML uses temporal links (TLINKs) (Setzer, 2002), to represent ordering. A TLINK expresses the temporal relation between two events. For example, an event *e1* can occur *before* another event *e2*. TimeBank is annotated using TLINKs, but the number of possible TLINKs in a document is large (quadratic in number of events). So annotation is restricted to a subset of TLINKs, leading to sparsity. To combat this, several works attempted to create denser corpora (Bramsen et al., 2006; Kolomiyets et al., 2012; Do et al., 2012; Cassidy et al., 2014), but still focused largely on local TLINKs.

Reimers et al. (2016) addressed high annotation cost by proposing a new scheme in which events were associated with explicit time expressions. Annotation effort now scaled linearly with number of events, making it feasible to annotate all of them. Using this scheme, they created EventTime, which had some discourse-level temporal annotation. However this dataset had one major drawback: events which could not be associated with a time expression were ignored. We observed that it may not always be possible to determine specific times for an event, but ordering it with respect to other events is often possible based on world knowledge. For example, consider the snippet: "Police discover body of *kidnapped* man. Police found the man's *dismembered* body wrapped in garbage bags". In this text, *dismembered* cannot be associated with a time. But the temporal relation between *dismembered* and *kidnapped* is clear because the kidnapping should have happened *before* dismembering. Based on this, we address the drawback in EventTime, by using TLINK-based annotation, which is expensive but allows more expressive power. Following TimeML, we augment

TimeBank-Dense (Cassidy et al., 2014) with global discourse-level TLINKs. To optimize manual effort, we automatically generate all TLINKs that can be inferred from EventTime. Then, we manually annotate a large subset of missing TLINKs involving events not associated with specific dates.

Most recently, Ning et al. (2018b) proposed a new scheme, which labels TLINKs based only on event start time. This improved inter-annotator agreement allowing for crowdsourcing of long-distance annotations at lower cost. However, they focused only on verb events, whereas our work is broader in scope and poses no such restrictions.

4.2.2 Temporal Ordering Systems

TimeBank and the TempEval tasks (Verhagen et al., 2007, 2010; UzZaman et al., 2013) spurred the development of many temporal ordering systems (UzZaman and Allen, 2010; Llorens et al., 2010; Strötgen and Gertz, 2010; Chang and Manning, 2012; Chambers, 2013; Bethard, 2013a). More recently, TimeBank-Dense and EventTime prompted development of newer models (Chambers et al., 2014; Mirza and Tonelli, 2016; Cheng and Miyao, 2017; Reimers et al., 2018). Most systems built for TimeBank/ TimeBank-Dense focus on TLINKs between events in the same or adjacent sentences, relying on local features rather than document-level structure, with some exceptions. Chambers and Jurafsky (2008); Denis and Muller (2011); Ning et al. (2017) introduce document-level consistency via integer linear programming constraints. Bramsen et al. (2006); Do et al. (2012) also incorporate document-level structure, but focus on different corpora. Reimers et al. (2018) develop a model for EventTime, which uses a decision tree of CNNs to associate each event from a document with a time. Several works have explored techniques to incorporate document-level cues such as event coreference (Do et al., 2012; Llorens et al., 2015) and causality (Do et al., 2012; Ning et al., 2018a) in temporal ordering systems. However, due to a lack of standard datasets focusing on global discourse-level links, most work has been evaluated on datasets of their own creation or standard datasets with mainly local TLINKs. This further stresses the need for a standardized benchmarking effort on discourse-level links, which we address by evaluating adaptations of several state-of-the-art systems on TDDiscourse (§4.4).

In addition to benchmarking SOTA systems, we develop a new joint BiLSTM+ILP model architecture, which extends the dependency parse-based BiLSTM model of Cheng and Miyao (2017) by introducing transitivity and semantic information extracted by a rule-based temporal parser called STAGE as ILP constraints. Prior work on incorporating information via ILP constraints typically added these constraints during postprocessing (Chambers and Jurafsky, 2008; Denis and Muller, 2011) or incorporated them during training for simpler classifiers like perceptrons (Ning et al., 2017). Conversely, in our formulation, ILP constraints are incorporated in a neural architecture during model training via a structured support vector machine (SSVM) framework. Our model formulation is close to the joint event-temporal model developed by Han et al. (2019), but we do not model event extraction. Instead, we introduce rich semantic information extracted by

STAGE.

4.2.3 Overview of Relevant Temporal Frameworks

To facilitate a better understanding of the STAGE temporal parser described later in the chapter, we briefly discuss some papers relevant to the temporal framework used by STAGE. Note that this is not an exhaustive review of the body of work on developing formal semantic frameworks to represent time, which is vast. Some foundational work that informed the development of STAGE is the framework by Allen and Hayes (Allen, 1984; Allen and Hayes, 1985; Allen, 1991). Allen and Hayes (1985) present an axiomatic model of time that expresses time spans as *intervals* or *moments*, distinguished by whether these time spans can be broken into smaller constituents or not. Most subsequent work on temporal semantics, including STAGE, maintains this influential distinction. The STAGE framework builds most directly on the OWL-S ontology (Pan and Hobbs, 2004), though it shares similarities with others such as Verhagen et al. (2005). Like Pan and Hobbs (2004), it identifies as possible time expressions *instants* and *intervals*, which represent moments along a timeline and spans of time, respectively. It also adds *ranges*, which cover spans of time like intervals, but reference the outer bounds of when the event takes place.

With advances in statistical learning, the field has been slowly shifting its focus away from formal models of semantics, though there have been periodic resurgences and some continuing work in adjacent fields. Some early contemporary formalizations of time can be found in the TimeML annotation scheme (Pustejovsky et al., 2003a) and subsequently developed corpora such as TimeBank (Pustejovsky et al., 2003b). Shared tasks using TimeBank data such as the TempEval 1-3 tasks (Verhagen et al., 2007, 2010; UzZaman et al., 2013) also motivated much recent work on temporal frameworks and taggers, such as HeidelTime (Strötgen and Gertz, 2010) SUTime (Chang and Manning, 2012), and TARSQI, which builds on Verhagen et al. (2005). As with most NLP tasks, a lot of this work focuses on news narratives, with the exception of the Temporal Event Ontology designed by Li et al. (2020) that specifically aimed to resolve complex temporal reasoning in clinical texts.

4.3 Dataset Creation

To address the lack of research on ordering long-distance event pairs, we develop TDDiscourse, the first dataset which focuses *explicitly* on annotating temporal relations (TLINKs) between event pairs that are more than one sentence apart. Additionally to maintain macro-level consistency, we create TDDiscourse by augmenting a subset of documents from an existing benchmark (TimeBank-Dense) with global TLINKs. Using the same set of 36 documents as TimeBank-Dense (Cassidy et al., 2014) and EventTime (Reimers et al., 2016) also facilitates comparison with previous work. Lastly, we utilize the same set of temporal relations as TimeBank-Dense, with the exception of the "vague"

Symbol	Relation
a	*e1* occurs **after** *e2*
b	*e1* occurs **before** *e2*
s	*e1* and *e2* are **simultaneous**
i	*e1* **includes** *e2*
ii	*e1* **is included** in *e2*

Table 4.1: Temporal relation set used in TDDiscourse. All relations are mutually exclusive.

label, since we do not require annotators to label all event pairs. Table 4.1 gives a brief summary of these relations. To add discourse-level links, we use two approaches:

- **Automatic inference:** We use a heuristic algorithm to automatically label global TLINKs using EventTime (§4.3.1) annotations, to generate a large number of links at low cost.
- **Manual annotation:** We manually label a subset of global TLINKs using document cues, world knowledge and causality (§4.3.2). To optimize human effort, we ensure that these TLINKs are not automatically inferable in the previous step.

4.3.1 Automatic Inference

This approach uses automatic inference to derive new TLINKs at low cost from EventTime (Reimers et al., 2016), which assigns specific times to events. EventTime divides events into two types: SingleDay and MultiDay. SingleDay events are assigned dates, while MultiDay events are assigned intervals. Possible event pairs can be divided into three categories: SS (both events are SingleDay), SM (one event is SingleDay while the other is MultiDay) and MM (both events are MultiDay). Not all assigned dates and intervals are exact. EventTime relies heavily on under-specified temporal expressions (such as "after1998-06-08"), making automatic inference non-trivial.

We follow separate algorithms to infer TLINKs for each pair type (SS, SM and MM). For SS pairs, both events are associated with dates, which may be expressed in one of four ways: MM-DD-YYYY, afterMM-DD-YYYY, beforeMM-DD-YYYY, afterMM-DD-YYYYbeforeMM-DD-YYYY, where MM-DD-YYYY stands for a specific date value. This results in 16 date combinations for SS links. We develop heuristics for each combination, which generate a temporal relation based on date values. Sample heuristics for 3 combinations are provided in Table 4.2. We develop similar rules for the remaining 13 cases, as well as for SM and MM links.

S1 Date Type	S2 Date Type	Procedure
MM-DD-YYYY	afterMM-DD-YYYY	• Get the relation (rel) between the date values from S1 and S2 • If rel is simultaneous or before, the SS link value is before • Else skip this link
MM-DD-YYYY	beforeMM-DD-YYYY	• Get the relation (rel) between the date values from S1 and S2 • If rel is simultaneous or after, the SS link value is after • Else skip this link
MM-DD-YYYY	afterMM-DD-YYYY beforeMM-DD-YYYY	• From S2, the date associated with after is named date1 and the date associated with before is named date2 • Get the relation (rel1) between date value from S1 and date1 from S2 • If rel1 is simultaneous or before, the SS link value is before • Get the relation (rel2) between date value from S1 and date2 from S2 • If rel2 is simultaneous or after, the SS link value is after • Else skip this link

Table 4.2: Sample heuristics for three SS link date combinations. Assume S1 and S2 indicate the points associated with events 1 and 2 which are to be linked.

Our heuristics were developed with a focus on precision to avoid adding incorrect links. Often, a relation cannot be generated. For example, consider two events associated with the same date "after02-01-1999". We know that both events occur after 02-01-1999, but we cannot infer their order with respect to *each other*. In such cases, we do not label the pair. For SM pairs, one event is associated with a time interval having begin and end dates. Here we use the SS pair inference algorithm to generate relations between the SingleDay event date and the MultiDay event begin and end dates. These relations are compared to infer the label for the pair. For MM pairs, both events have begin and end dates. We infer relations between begin and end points using SS link inference and use these to infer the pair label. After inference, we perform temporal closure, according to Chambers et al. (2014). To evaluate validity of generated TLINKs, we randomly sample a subset of 100 TLINKs and ask three annotators to determine the correctness of the labels. Annotators were volunteers with no vested interest in the corpus. All annotators unanimously agree with the assigned label in 99% cases. We call this subset **TDD-Auto**.

4.3.2 Manual Annotation

In this phase, we ask expert annotators, with a background in computational linguistics, to label discourse-level TLINKs that cannot be inferred automatically. Getting expert annotation for all missing TLINKs is expensive. Hence, we randomly subsample a set of TLINKs not annotated by TimeBank-Dense or automatic inference. This subset is as large as TimeBank-Dense, thus doubling the data size while making the overall task harder (see §4.4). Note that TLINKs annotated in this phase may involve events for which a specific time of occurrence cannot be determined, which were ignored in EventTime. We refer to this subset as **TDD-Man**.

Since TLINKs are not restricted to the same or adjacent sentences, our annotation task becomes harder, requiring cues from the entire document. Many TLINKs also require the use of causal links and world knowledge to label the relation. Based on our observations, we carefully develop a detailed coding scheme. To ensure high inter-annotator agreement, we refine our scheme over multiple rounds of annotation and discussion of disagreements.

Coding Scheme

Our scheme reduces the task of labeling a TLINK to a set of concrete decision steps:

1. Using textual cues
2. Using world knowledge
3. Using narrative ordering

A TLINK may be assigned a label at any step. If it cannot be assigned a label, it moves on to the next step. Information from previous steps is retained, making it possible to combine multiple sources of evidence. For example, textual cues may not suffice, but they can be used in conjunction with world knowledge to label a pair. We choose to organize our coding scheme as mentioned above, to make the process of gathering evidence about an event pair systematic, and ensure that experts

Snippet
Atlanta nineteen ninety-six. A bomb **blast shocks** the Olympic games. One person is **killed**. January nineteen ninety-seven. Atlanta again. This time a **bomb** at an abortion clinic. More people are **hurt**.
Event pair: *blast*, *hurt*
Relation: before
Textual cues: Event *blast* occurred in 1996. Event *hurt* occurred because of second bomb blast in 1997.

Table 4.3: Sample document-level textual cues used during temporal annotation.

do not miss important cues. The final step is guaranteed to assign a label. We choose not to allow annotators to leave event pairs unlabeled or label them "vague", to keep them from overusing this option. Owing to this decision, we need to develop mechanisms for handling TLINLKs containing events which have not actually occurred (eg: negated, hypothetical or conditional events). Drawing from prior work, we interpret these events using a *possible worlds* analysis, in which the event is treated as if it has occurred. We refer interested readers to Chambers et al. (2014) for a more detailed discussion.

Using textual cues

In this step, we use document-level textual cues to label a TLINK. These cues used are generally similar to those used in previous datasets (Cassidy et al., 2014). Table 4.3 gives an example of the types of cues used.

A key textual cue we use in this step is event coreference. Traditionally, event coreference has not been used for temporal annotation because the occurrence of coreferent events in adjacent sentences is rare. However, this cue is crucial for global discourse-level annotation. Since TimeBank-Dense does not contain event coreference information, we develop an additional procedure to annotate the documents for event coreference. Our procedure is based on the ERE (Entities, Relations, and Events) scheme (Song et al., 2015), which cannot be directly used for TimeBank due to differing notions of what constitutes an event and different metadata. In our procedure, events are considered coreferent *iff* they share the following:

- Entities involved in the event
- Temporal attributes
- Location attributes
- Realis (whether event is real or hypothetical)

Events which are synonymous in context are also considered coreferent (for instance, in "...held an interview Monday. The segment covered...", *interview* and *segment* are synonymous). These attributes (barring temporal) are not provided in TimeBank and must be inferred. Often, an event

Snippet
Their talks have been **bedeviled** by a number of **disputes**. **Event pair:** *disputes*, *bedeviled*
Coreferent: Yes
Reasoning: The event *disputes* is itself the entity enacting the event *bedeviled*. The events take place over the same time period and location, and are both real events. Thus, we can conclude the events are coreferent.
Snippet
Lower rates have **helped** invigorate housing by **making** loans more affordable. **Event pair:** *helped*, *making*
Coreferent: No
Reasoning: Though the events share an agent ("lower rates") and realis states, they act on different patient entities and thus are not coreferent.

Table 4.4: Sample coreferent and non-coreferent event pairs from TimeBank-Dense.

may only have partial information about these attributes - here we use human judgment. Our definition of coreference is closer to the strict notion of "event identity" in Light ERE than the relaxed definition in Rich ERE. Table 4.4 provides some examples of coreferent and non-coreferent event pairs from TimeBank-Dense as per our coding procedure.

To test our procedure, we select all "simultaneous" TLINKs from TimeBank-Dense to ensure that our sample contains a sizeable proportion of *possibly* coreferent event pairs. The corpus contains 179 "simultaneous" links, of which 93 are event pair TLINKs. Our first annotation pass achieves high agreement between two annotators, with a Kappa of 0.70. We refine our guidelines through an adjudication step, reaching perfect agreement on this sample. Post-adjudication guidelines are used to annotate event coreference for all documents. Resulting annotations are used as textual cues in our temporal annotation scheme. Based on textual cues, an appropriate label from Table 4.1 is assigned to a TLINK. Coreferent TLINKs are labeled "simultaneous". Unlabeled links move on to the next decision step.

Using world knowledge

This step uses real world knowledge to determine causal/prerequisite links which are used to label a TLINK. We consider both events in the TLINK and determine whether they possess one or both of the following:

- **Causal Link:** Two events have a causal link if the occurrence of one event results in the other event coming about. For example, in the sentence "The paper got wet when I spilled water on it", the event pair (spilled, wet) have a causal link.

Rule	Label
TLINK=(A, B), A=P	Before
TLINK=(A, B), A=I	Includes
TLINK=(B, A), A=P	After
TLINK=(B, A), A=I	Is Included

Table 4.5: Labels assigned to event pairs based on event and TLINK metadata.

- **Prerequisite Link:** Two events have a prerequisite link if one event *must* occur before the other can happen. For example, in the sentence "We cooked dinner and ate it", the event pair (cooked, ate) have a prerequisite link. Note that we use the knowledge that a meal must be cooked before it can be eaten, though it is not explicitly mentioned.

We examine the event pair in the context of the entire document to detect causal/prerequisite links, also allowing weak or transitive links. For instance, in the text "Diplomacy is making headway in resolving the UN's standoff with Iraq. One major sticking point has been Iraq's proposal...", *proposal* causes *standoff*, which is a prerequisite for *resolving*. Hence, the pair (proposal, resolving) is considered causal/prerequisite. Our assignment of causal/prerequisite links is unordered. For example, reverse event pairs (wet, spilled), (ate, cooked), and (resolving, proposal) are also considered causal/prerequisite. Link order is taken into consideration while assigning a temporal relation.

If two events contain a causal/prerequisite link, we identify the event in the pair that causes or is a prerequisite for the other. We call this event "A" and the other "B". For example, (spilled, wet) is expressed as (A, B), while (wet, spilled) is expressed as (B, A). To label the TLINK, we determine whether A is a point (P) or interval (I) event using existing date annotations from EventTime (Reimers et al., 2016). This helps us catch cases where A is a long-lasting interval and the time span for B is completely included in A. For instance, in "the war forced civilians to evacuate", (war, evacuate) has a causal/prerequisite link with *war* being event A. Though *war* caused *evacuation*, it is reasonable to expect that the war started *before* and ended *after* evacuation. If A is not present in EventTime (i.e it cannot be assigned a specific time), we use our judgment to determine event length. We then assign a label as per Table 4.5. Unlabeled links are passed to the next step.

Using narrative ordering

This step uses a heuristic based on the intuition that events in news narratives are often presented in chronological order. To label a TLINK, we determine which event appeared first in the document. This event is called "A", and the other is "B". We then detect whether A is a point (P) or interval (I) from EventTime, falling back to our own judgment if it is not present. Finally, a label is assigned following Table 4.5. This step is guaranteed to assign a label since every pair will have a narrative-based order.

Dataset	Kappa
TimeBank	0.71
TimeBank-Dense	0.56-0.64
TDD-Man	0.69

Table 4.6: Inter-annotator agreement (Cohen's Kappa) on temporal ordering datasets. Kappa scores for TDD-Man are reported on the test set containing 1500 links.

	a	b	s	i	ii
a	137	22	0	12	22
b	30	311	1	72	23
s	0	0	42	5	4
i	9	36	3	462	35
ii	12	32	0	21	209

Table 4.7: Relation agreement between annotators on the TDD-Man test set containing 1500 links. Here a, b, s, i, ii refer to the temporal relations "after", "before", "simultaneous", "includes", and "is included".

Inter-annotator agreement

Our annotation scheme was developed over multiple rounds of coding and discussion between two experts. In each round, experts separately annotated a set of 10-15 TLINKs, sampled from documents in the development set. Cohen's Kappa was computed and disagreements were discussed. TLINKs were changed in every round to ensure exposure to diverse event pair types. Inter-annotator agreement in preliminary rounds ranged from 0.48-0.69. The final coding scheme resulted in an agreement of 0.69 on the test set. Table 4.6 shows that our agreement is comparable to prior work. Table 4.7 presents a class-wise distribution of agreements between pairs of annotators. Disagreements mainly include cases where one annotator chose after/before while the second chose includes/is included (64%). This indicates that determining precise end-points for an interval event is difficult, as corroborated by Ning et al. (2018b).

4.3.3 Dataset Statistics

Our data construction pipeline produces the first dataset focused on temporal links between global discourse-level event pairs (**TDDiscourse**), consisting of two subsets **TDD-Man** and **TDD-Auto**. Table 4.8 presents train, dev and test set sizes for both subsets, Timebank-Dense as well as an augmented version of TimeBank-Dense with additional links inferred via temporal closure. Our complete dataset is 7x larger than both, indicating that our construction adds valuable new TLINKs. **TDD-Man** itself is as large as TimeBank-Dense and can be used in isolation, however incorporating **TDD-Auto** provides a large amount of silver training data making the task more amenable to deep neural net approaches. Note that Appendix C presents some examples of annotated instances from

Dataset	Train	Dev	Test
TB-Dense	4032	629	1427
TB-Dense + Closure	4399	722	1575
TDD-Man	4000	650	1500
TDD-Auto	32609	1435	4258

Table 4.8: Dataset sizes for TimeBank-Dense and our dataset. Note that we only count event-event TLINKs.

both TimeBank-Dense and TDDiscourse.

Table 4.9 presents class distributions for TDD-Man and TDD-Auto test sets. Though there is a clear majority class, both sets are more balanced than TimeBank-Dense, in which 40% event pairs are labeled "vague". To evaluate the presence of long-distance TLINKs, we present the distribution of distance between event pairs from annotated TLINKs in Table 4.10 which shows that nearly 53% TLINKs in our dataset comprise of event pairs which are more than 5 sentences apart. Further, to gain deeper insight into global discourse-level phenomena exhibited by our dataset, we augment 3 documents from the test set (107 manual and 110 automated event pairs) with additional annotations about phenomena required to label them correctly. We consider the following phenomena:

- **SingleSent (SS):** Textual cues from sentences containing the events suffice to predict the relation (irrespective of distance).
- **Chain Reasoning (CR):** Correct relation prediction requires reasoning about other events from the document.
- **Tense Indicator (TI):** For verb events, tense information indicates the correct relation.
- **Future Events (FE):** One or both events from the pair will occur in the future.
- **Hypothetical/ Negated (HN):** One or both events are hypothetical or negated.
- **Event Coreference (EC):** Event coreference resolution is needed to predict relation.
- **Causal/ Prereq (CP):** Causal/ prerequisite links must be identified to predict relation.
- **World Knowledge (WK):** Real world knowledge is needed to identify the relation.

Table 4.11 shows the distribution of these phenomena in TDD-Man and TDD-Auto. TDD-Man shows a higher percentage of difficult phenomena (CR, CP). On the other hand, TDD-Auto shows high prevalence of SS, indicating that local information may be sufficient to label many long-distance links in this subset correctly. This principled comparison of both subsets leads us to hypothesize that models which perform well on TimeBank-Dense, should achieve similar scores on TDD-Auto but perform much worse on TDD-Man.

4.4 Benchmarking State-of-the-Art Models

4.4.1 Model Details

To statistically evaluate and establish the difficulty of TDDiscourse, we benchmark and study the performance of four models, which have achieved SOTA performance on TimeBank-Dense.

Dataset	a	b	s	i	ii
TB-Dense	0.18	0.22	0.02	0.05	0.06
TDD-Man	0.13	0.27	0.03	0.38	0.19
TDD-Auto	0.28	0.32	0.16	0.11	0.13

Table 4.9: Class distributions for our test sets and TimeBank-Dense. Note that the distribution for TimeBank-Dense does not sum to 1, since it includes a vague class.

Dataset	<5	<10	<15	<20	>20
TDD-Man	0.40	0.40	0.15	0.04	0.01
TDD-Auto	0.50	0.32	0.12	0.05	0.01

Table 4.10: Distribution of distance between events for all TLINKs in our test sets (in terms of #sentences).

Three of these models (CAEVO, BiLSTM, SP) are non-transitive and make separate local decisions for each TLINK, which may result in global inconsistency. For example, for events A, B and C, if A occurs before B and B occurs before C, transitivity implies that A occurs before C. Models classifying each pair independently may assign a different relation to A-C. The fourth model (SP+ILP) attempts to correct for this by incorporating transitivity rules as integer linear programming (ILP) constraints into the perceptron model (SP). The SP+ILP model was initially proposed by Ning et al. (2017), but we use a different ILP formulation to impose stricter transitivity constraints in order to improve tractability on our data, which contains 7x more TLINKs. In addition, we make minor modifications to the non-transitive SOTA temporal ordering models so that they are better equipped to handle discourse-level TLINKs. Following is a brief description of all four models, along with our additional modifications:

- **CAEVO** (Chambers et al., 2014): This system consists of a series of specialized learners

Phenomenon	TDDMan	TDDAuto
SS	25.23%	90.91%
CR	58.88%	9.09%
TI	12.10%	46.36%
FE	36.45%	29.09%
HN	14.02%	19.09%
EC	16.82%	4.55%
CP	64.49%	29.09%
WK	16.82%	0.91%

Table 4.11: Distribution of various phenomena in the annotated test subset. These phenomena were labeled manually.

(sieves) which include heuristic rules and trained models for temporal relation prediction. For each document, sieves run in decreasing order of precision. Decisions made by earlier sieves constrain following ones. This framework integrates transitive reasoning, but decisions made by earlier sieves cannot be overturned, causing error cascades. To extend CAEVO, we increase window sizes and remove the AllVague sieve, since our data does not include the vague class. We also remove the WordNet sieve and add MLEventEventDiffSent. For more details on these sieves, we refer interested readers to Chambers et al. (2014).

- **BiLSTM** (Cheng and Miyao, 2017): Inspired by Xu et al. (2015), this model uses a BiLSTM classifier. For each event pair, dependency paths from both events to the sentence root are fed to a BiLSTM. For events in adjacent sentences, both event sentences are assumed to be connected to a "common root". We follow the same framework to build a BiLSTM.

- **SP** (Ning et al., 2017): A baseline structured perceptron model which uses handcrafted features such as event text, part-of-speech, other metadata such as modality and tense, etc. to compute representations for each event from the event pair, followed by a perceptron model to predict the temporal relationship.

- **SP+ILP** (Ning et al., 2017): This model incorporates transitivity constraints via ILP into the perceptron (SP), explicitly enforcing global consistency. This model was originally trained on TimeBank-Dense which contains fewer TLINKs per document, making joint learning tractable with their loose transitivity constraints. But loose transitivity is an issue for our data with 7x more TLINKs, since the number of constraints increases tremendously. To improve tractability, we define a stricter transitivity constraint. Let E, R and P be sets of events, temporal relations and event pairs respectively($P = \{(e_i, e_j) \in E \times E | e_i, e_j \in E, i \neq j\}$). We define an array of binary indicator variables y, where $y_{<r,i,j>}$ indicates whether the relation r holds between events e_i and e_j. Our objective function is defined as:

$$\arg\min_{y} \sum_{<e_i,e_j>\in P} \sum_{r \in R} -y_{<r,i,j>} \log p_{<r,i,j>} \tag{4.1}$$

subject to the following constraints:

$$y_{<r,i,j>} \in \{0,1\}, \forall (e_i, e_j) \in P, \forall r \in R \tag{4.2}$$

$$\sum_{r \in R} y_{<r,i,j>} = 1, \forall (e_i, e_j) \in P \tag{4.3}$$

$$\begin{gathered} y_{<r1,i,j>} + y_{<r2,j,k>} - y_{<r3,i,k>} \leq 1, \\ \forall (e_i, e_j), (e_j, e_k), (e_i, e_k) \in P, \forall (r1, r2, r3) \in TC \end{gathered} \tag{4.4}$$

where $p_{<r,i,j>}$ is the probability that event pair (e_i, e_j) has label r. (4.2) ensures that indicator variables are binary, (4.3) forces event pairs to be assigned a unique label and (4.4)

System	TB-Dense			TDD-Auto			TDD-Man		
	P	R	F1	P	R	F1	P	R	F1
MAJOR	40.5	40.5	40.5	34.2	32.3	33.2	**37.8**	**36.3**	**37.1**
CAEVO	49.9	46.6	48.2	61.1	32.6	42.5	32.3	10.7	16.1
BiLSTM	63.9	38.9	48.4	**55.7**	**48.3**	**51.8**	24.9	23.8	24.3
SP	37.7	37.8	37.7	43.2	43.2	43.2	22.7	22.7	22.7
SP+ILP	**58.4**	**58.4**	**58.4**	46.4	45.9	46.1	23.9	23.8	23.8

Table 4.12: Performance of SOTA models on TB-Dense, TDD-Auto and TDD-Man. MAJOR represents a majority-class baseline. We report performance on non-vague event-event links for TB-Dense to ensure fair comparison.

imposes transitivity. TC denotes the set of transitive relation triples.[2] Relation probabilities ($p_{<r,i,j>}$) come from the structured perceptron. Evaluating both this model and the structured perceptron (**SP**) lets us study the effect of introducing global consistency via ILP.

4.4.2 Results

We evaluate the performance of 4 models (**CAEVO**, **BiLSTM**, **SP** and **SP+ILP**) on **TDD-Auto** and **TDD-Man**. **SP** is a perceptron-based classifier, while **SP+ILP** introduces transitivity via ILP into the perceptron, which allows us to evaluate the impact of introducing transitivity constraints. For tractability, we limit all models to predicting temporal relations for event pairs which are 15 or fewer sentences apart. This discards only 5% of our data (Table 4.10). Table 4.12 presents the results. We also include model performance on the TimeBank-Dense dataset (**TB-Dense**) to demonstrate that our additional modifications do not affect performance on local TLINKs.

All models perform better than a majority class baseline on TDD-Auto. The BiLSTM and SP perform particularly well, achieving a higher F1 than TB-Dense, while CAEVO and SP+ILP show slight degradation in comparison to TB-Dense. This corroborates our hypothesis that many long-distance TLINKs in TDD-Auto can be handled with local sentence-level information. However, all models show a significant drop on TDD-Man, with none outperforming a majority class baseline, indicating that this dataset contains a higher proportion of complex discourse-level temporal ordering phenomena. Finally, we observe that SP+ILP outperforms SP across all datasets, indicating that transitivity constraints improve overall performance. We perform further analyses of global consistency and model errors, which offer valuable insights into phenomena which are not handled by current models, posing interesting challenges for future work and further highlighting the challenging nature of our new benchmark dataset.

[2] ("before", "before", "before") form a transitive relation triple as A before B and B before C implies A before C

Error Category	%Cases
WK	40
HN	31
ES	22

Table 4.13: Proportion of TDD-Man cases falling into various error categories. Note that WK, HN and ES refer to the "World Knowledge", "Hypothetical/Negated", and "Event Structure" error categories described in Section 4.5.2.

4.5 Analyzing State-of-the-Art Model Performance

4.5.1 Evaluating Global Consistency

As mentioned earlier, most SOTA models make separate local decisions for each pair which may not be globally consistent. Adding global consistency via ILP improves the overall performance of a local classifier, as evinced by F1 gains observed on adding ILP to SP. We further validate this observation by specifically evaluating predictions for global consistency through a transitivity analysis of SP+ILP and BiLSTM, which is the best-performing model, on TDD-Auto. For this analysis, we go through all possible event triples (e_1, e_2, e_3). For each model, if (e_1, e_2), (e_2, e_3) and (e_1, e_3) are all assigned temporal labels, we check whether label assignments are consistent. For example, e_1 after e_2, e_2 after e_3 and e_1 after e_3 is a consistent assignment. We observe that though the BiLSTM has higher F1, it maintains transitivity in 41.9% cases, while SP+ILP enforces transitivity in 53.6% cases, which is a 12% increase. This indicates that incorporating such constraints into models introduces an inductive bias which makes predictions consistent with human expectations.

4.5.2 Error Analysis on TDD-Man

The dismal performance of all models on TDD-Man indicates that this subset contains several temporal phenomena that SOTA models, tailored to do well on head cases, fail on. To identify these phenomena, we manually look at 100 event pairs from TDD-Man for which all models predicted the incorrect label. For each case, we label it for the presence of one or more of the following major causes of error:

- **World Knowledge (WK):** This includes cases which require models to have knowledge about typical event duration/ordering (e.g., *war* is a long-term event), as well as lexical entailment (e.g., *military actions* could refer to the same event as *air strikes*).
- **Event Structure (ES):** This includes cases which require models to handle complex event structure such as event coreference, sub-events, and aspectual prediction, which is a gram-

matical device that focuses on different facets of event history (e.g., using "begin" to indicate initiation of an event).

- **Hypothetical/Negated events (HN):** This includes cases which require models to handle hypothetical events, which may or may not occur, and negated events, which have definitely not occurred. These cases are not exclusive to TDD-Man, but are also present in both TB-Dense and TDD-Auto.

Table 4.13 shows the proportion of cases that fall into each of these error categories. Note that these categories are not exclusive, and one event pair may fall into multiple categories. World knowledge is the largest source of errors, indicating that incorporating commonsense knowledge about typical event duration and ordering into these models is a key direction for future research. Indeed, some efforts to automatically construct resources containing temporal commonsense knowledge have already been made (Zhou et al., 2020).

4.6 Case Study: Adapting From Local to Long-Distance Event Ordering

After developing and validating TDDiscourse, we use this dataset as a testbed to study micro-level adaptation. As described earlier, we do this by training a model on local event pairs from the TimeBank-Dense dataset, and evaluating its performance on long-distance event pairs from TDDiscourse. Since TimeBank-Dense and TDDiscourse both use the same set of 36 documents, all macro-level dimensions are consistent across the two datasets. Any differences between the two would primarily arise from variation in linguistic phenomena present in the datasets due to a difference in task focus (local vs long-distance pairs). Given this unsupervised micro-level adaptation setting, we evaluate the performance of a baseline task model (with no adaptation), a data-centric adaptation method from the pseudo-labeling fine category and a model-centric adaptation method from the loss augmentation fine category. Following subsections describe the task model and adaptation methods in more detail.

4.6.1 Baseline Task Model Architecture

Figure 4.1 gives a brief overview of the architecture of the model chosen as our baseline task model (BiLSTM), which is a re-implementation of the dependency parse-based BiLSTM model developed by Cheng and Miyao (2017). As described earlier, this model, which is one of the state-of-the-art models on TimeBank-Dense, works by using a BiLSTM to construct representations of dependency paths from source and target events to the sentence root (or "common root", if source and target events are in different sentences). These representations are then fed to an MLP to predict the temporal relation. From our benchmarking experiment on TDDiscourse, we observed that BiLSTM was the best-performing model on TDD-Auto, and the best-performing model on TDD-Man out

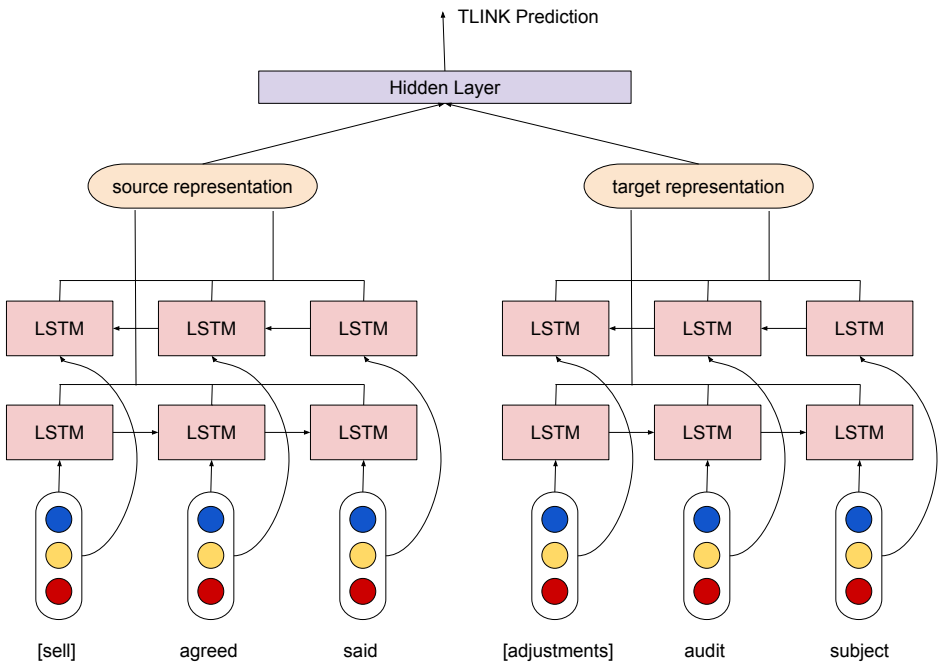


Figure 4.1: Architecture of the dependency parse-BiLSTM model used as the temporal ordering task model for our micro-level adaptation case study.

of all SOTA models. Note that no SOTA model could beat majority class baseline on TDD-Man. Motivated by its high performance on TDDiscourse, we chose the BiLSTM model as our baseline task model.

4.6.2 Joint BiLSTM+ILP Architecture: A Loss Augmentation Adaptation Method

To improve unsupervised micro-level adaptation, we propose a joint BiLSTM+ILP model architecture that can incorporate pre-defined heuristics (e.g., transitivity) into a neural model during its training process. We achieve this by formulating integer linear programming (ILP) constraints to represent chosen heuristics, which are then incorporated into neural model training using a structured support vector machine (SSVM) framework. Since this method works by editing the loss function, it falls into the loss augmentation fine category of model-centric adaptation methods.

To understand the SSVM framework better, consider that we want to incorporate the property of transitivity into the BiLSTM model using ILP constraints. We use the same ILP formulation of transitivity as described in §4.4.1. For all event pairs in a document, we first use the BiLSTM to compute relation probabilities. Using these scores, we solve the ILP optimization and obtain a set of predictions y. Given gold predictions y' and BiLSTM probablities p, the SSVM framework

computes a structured hinge loss using the following formulation:

$$L(y, y') = \max(0, \Delta(y, y') + \Psi(y, p) - \Psi(y', p)) \tag{4.5}$$

Here $\Delta(y, y')$ is a distance measure between the gold and predicted labels. We use Hamming distance in our formulation. $\Psi(y, p)$ and $\Psi(y', p)$ are scoring functions used to compute scores for the gold and predicted labels. We use the same function as the ILP objective (without the negative sign) for score computation. The main intuition behind the hinge loss formulation is that if the gold labels y' are not scored higher than the predicted ones y (with a margin of $\Delta(y, y')$), there will be a non-zero loss. The objective is to minimize this margin loss. This framework allows the gradient updates during BiLSTM training to be influenced by the ILP optimization process, thereby infusing some knowledge of transitivity into the model, instead of just using ILP as a post-hoc step to maintain transitivity. Other heuristics aside from transitivity are also incorporated in a similar manner. The ILP objective/constraint formulation are edited to reflect these heuristics, but the underlying SSVM framework remains unchanged.

For this case study, we conduct experiments incorporating two types of heuristics: (i) transitivity, and (ii) semantic information extracted from explicit time cues. Transitivity is a key property of temporality, and has proved to be extremely helpful in improving the accuracy and consistency of temporal ordering models, as also demonstrated by our benchmarking experiment on TDDiscourse. On the other hand, incorporating explicit time cue information into temporal ordering models, particularly neural architectures, is an under-explored yet promising avenue to improve model performance and generalization. To obtain maximum utility from time cue information, we should be able to extract time cues and generate valid ILP constraints from them, with reasonably high precision. We use the STAGE (Semantic Temporal Alignment Grammatical Extraction) tool to achieve this. While not the main focus of our work, a basic understanding of STAGE is required to understand how it generates ILP constraints that can be introduced into our BiLSTM+ILP model, which is provided in the following overview.

4.6.3 Overview of STAGE: A Tool for Automated Time Cue Extraction

STAGE consists of two main parts: (i) a novel semantic framework to represent explicit time cues, and (ii) a tool to automatically extract these cues from raw text.

STAGE Semantic Framework

Several temporal logic frameworks and tools have been developed that automatically extract temporal information with good reliability and accomplish some level of semantic normalization. However, older ontologies, which are constructed by hand, guided in a top-down fashion by theoretical insights into language, provide limited coverage and do not scale well to current datasets.

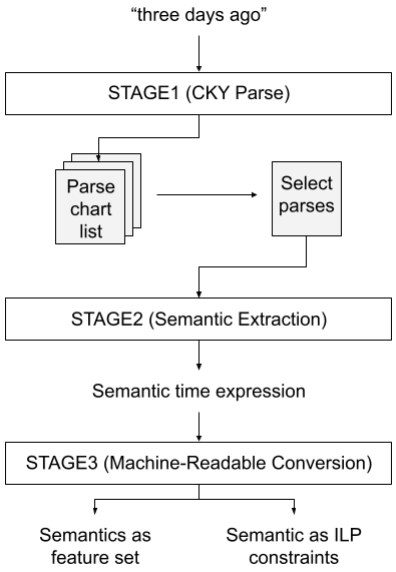


Figure 4.2: Overview of the three stage architecture of the STAGE extraction tool.

Conversely, recent approaches perform well on current datasets while sacrificing some rich semantic information that would be valuable in more rigorous temporal reasoning. The STAGE semantic framework is designed to balance both, maintaining a semantically rich, complex representation of time that is yet standardized enough that it can be extracted automatically from explicit textual time cues in large corpora.

In STAGE, an explicit textual time cue or "time expression" refers to a contiguous string of text that communicates a concept about time. Depending upon its contextualization, a time expression can be assigned to one (or more) of the three basic categories: instant, interval or range, as shown by the examples below:

- "The celebration took place *on January 1st, 2001*": *instant* occurring on 01/01/01.
- "People were waiting *from January to June*": *interval* starting in January and ending in June.
- "The party will happen *sometime in December*": *range* covering the month of December.
- "We should meet *for an hour sometime next week*": both an *interval* lasting one hour and a *range* covering the next week.

Beyond assignment of expressions to the categories enumerated above, and formalization of the status of temporal expressions not explicitly connected to an event in a discourse, STAGE also addresses the issue of comparison between temporal expressions. The goal is to design the STAGE temporal expression ontology in such a way that models can standardize time expressions into a common format, facilitating easy comparisons between time expressions. This approach makes the following specific modifications to the Pan and Hobbs (2004) framework:

1. Lengths of time are represented using a standard unit (hours) in order to facilitate comparison between semantic objects that may have been expressed in different units.

Text	Semantic type
"four hours"	A length of time.
"in four hours"	An instant with a clear position on a timeline.
"for four hours"	An interval with clear duration and vague position.
"within four hours"	A range with clear duration and position that an event occurs for some vague duration and position within.

Table 4.14: Impact of function words on semantic meaning of time expression.

2. Relative expressions (e.g., "three days ago") are converted to dates based on document date, when known.
3. Intervals/ranges are represented as one (or a combination) of the following properties: starting point, ending point, and length. This better mimics the ways in which humans describe time spans.
4. Instead of resolving relationships between time expressions in a rule-based manner as in previous temporal formalisms, each temporal expression is represented separately but the representation includes some associated properties that provide clues for uncovering the relationship between temporal expressions downstream. In this way, individual temporal expressions extracted by STAGE are somewhat more elaborate than in other recent work (see Table 4.16) in ways that support better performance on event ordering.

STAGE Extraction Tool

Building on the framework described in the previous section, STAGE also includes a semantic extraction tool focusing on the identification and arrangement of time expressions along a single standardized timeline. The tool utilizes lexical time cues alongside function words, which were frequently omitted from consideration in published annotation schemes for time expressions (e.g., TempEval-3 Platinum (UzZaman et al., 2013)). But function words have significant impact on the underlying semantic meaning of a time cue. Table 4.14 shows how distinct function words can change the properties and even type of the semantic representation. STAGE uses a newly-designed context-free semantic grammar to parse time expressions into representations according to the stable correspondence between function words and temporal concepts, such that the results have utility for tasks like event ordering that require comparison between temporal expressions.

STAGE does its extraction and representation work in three steps, separated into three distinct modules shown in Figure 4.2. The first module produces all potential parses for a time cue. This module takes a text string as input, identifies the words which belong to STAGE's temporal vocabulary, and applies the STAGE temporal grammar rules. It uses a binary CKY chart parser to efficiently generate all possible parses for each input string, and outputs the full chart. As an

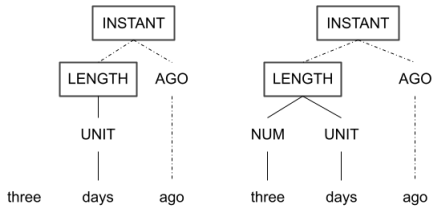


Figure 4.3: Example of first-step STAGE output.

Feature	Value
Is instant?	True if the expression represents a single instant
Is start interval?	True if the startpoint of the time expression is the start of the event, False if a lower bound on the start.
Is end interval?	True if the endpoint of the time expression is the end of the event, False if an upper bound on the end.
Is length interval?	True if the length given for the expression is the exact length of the event, False if an upper bound on the length.

Table 4.15: Features constructed by STAGE that can be integrated with neural temporal ordering models.

example, Figure 4.3 shows all parse trees produced for the time cue "three days ago". The trees that do not span the full time cue often resolve to complete (though less semantically specific) time expressions. The second module produces a logical representation of a text string's underlying semantic information using a set of heuristically-determined semantic rules. It takes as input a set of parse trees. STAGE typically chooses from the first module's output, the parse tree spanning the largest subsection of the input, which also resolves to one of the three "complete" expression types (instant, interval, or range). The nodes of the parse tree instruct the module how to apply the semantic transformations, and the output is a formal semantic representation of the original text string. In the example above, the module behaves as follows:

- **START:** "three days ago" $\longrightarrow$ NUM(val=3) UNIT(val=day) ago
- **RULE:** NUM + UNIT = LENGTH $\longrightarrow$ NUM(3) + UNIT(day) = LENGTH(num=3,unit=day)
- **RULE:** LENGTH + ago = INSTANT $\longrightarrow$ LENGTH(num=3,unit=day) + ago = INSTANT(anchor="present", dist from anchor=LENGTH(number=3, unit=day))

STAGE rules allow for complete time expressions to be transformed into other types with infinite recursion. If the string is changed to "before three days ago" we would see:

- **RULE:** before + INSTANT = INTERVAL $\longrightarrow$ before + INSTANT(anchor= ...unit=day) = INTERVAL(start=Unknown, end=Instant(anchor=...unit=day), length=Unknown)

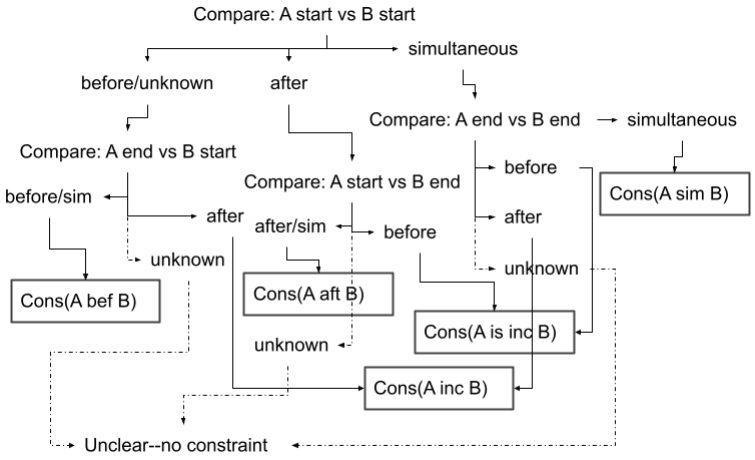


Figure 4.4: Flowchart detailing constraint logic used in STAGE.

Model	TBDense	TE3
HeidelTime	N/A	87.7
SCFG	N/A	81.6
SUTime	N/A	91.3
STAGE (=/+)	91.2	86.7
STAGE (+)	66.8	63.2

Table 4.16: Comparison of STAGE with other state-of-the-art parsers on temporal expression identification.

The final module takes this high-level logical representation and converts it to a machine-readable form for downstream tasks. STAGE is capable of converting information into two different machine-readable formats: (i) a set of input features, and (ii) constraints dictating the order of certain event pairs. For our case study, we primarily use the constraint representation, so we only discuss this format in further detail here.[3] To generate constraints, STAGE examines the start and end points for each event in the pair and heuristically identifies pairs for which the relation is certain based on these features alone. The constraint generated pushes the model to prioritize the predicted relation over others for this pair. See example of resulting constraint logic output shown in Figure 4.4.

For the example input string "three days ago", if our dataset included three events, where "three days ago" is linked to event A, and event B takes place "two days ago" while C is "one week ago", STAGE would produce constraints "A before B" and "A after C".

Evaluating STAGE

In addition to its ability to extract valid ILP constraints from explicit time cues, another reason for using the STAGE system in our case study is its high precision performance on identifying explicit

[3]Interested readers can refer to Breitfeller et al. (2021) for more details on the feature representation.

temporal expressions from documents. To evaluate this, we test STAGE on TempEval-3 Platinum (UzZaman et al., 2013), a benchmark dataset for temporal expression identification and compare its performance to SOTA time expression extractors such as HeidelTime (Strötgen and Gertz, 2010), SUTime (Chang and Manning, 2012) and the synchronous context-free grammar from Bethard (2013b). Table 4.16 shows the performance of all these systems. Model performance is measured using a *relaxed* string matching metric, where a time expression is considered to match the gold expression if it includes the full string along with additional words that do not change the meaning of the time expression (ex. if the gold annotation was "Monday" and the model output was "the Monday", this would be considered a match). We use this relaxed match metric for comparison because STAGE is intended to capture the specific way a time cue positions an event relative to each time point. This often results in extraction of longer spans of text including function words, which are typically ignored in gold annotations. In addition to "relaxed match" performance ("STAGE (=/+)"), we also highlight the proportion of cases in which the STAGE output produces a longer time expression than gold annotation ("STAGE (+)"). This indicates that for a large number of cases, STAGE is able to extract richer temporal information as compared to other SOTA parsers. Following are some examples from a qualitative analysis that highlight this richness in expressions extracted by STAGE vs gold annotations:

- Gold annotation "December" vs STAGE output "in December": builds a possible range in December the event must take place within
- Gold annotation "the fourth quarter" vs STAGE output "for the fourth quarter" expresses an event lasting the entire quarter
- Gold annotation "the day" vs STAGE output "later in the day", which identifies and links event to a sub-section of the full day

In addition to benchmarking on TempEval-3 Platinum, we also evaluate the performance of STAGE on TimeBank-Dense. The performance scores indicate that STAGE can identify temporal information with very high precision on this dataset, motivating its utility for our case study.

4.6.4 Adding STAGE Constraints to BiLSTM+ILP

Figure 4.5 gives a brief overview of the integrated model pipeline after incorporating STAGE time cue extraction and constraint generation into the BiLSTM+ILP model. To incorporate STAGE-generated constraints into the ILP formulation, we first add dummy events representing the time expressions that have been extracted by STAGE to the ILP. Let this set of dummy events be E_d. The ILP now contains new variables for each pair of events (e_i, e_j) where e_i, e_j or both are dummy events from E_d, and the non-dummy event is from the set E. For each date in E_d, STAGE generates temporal relations between the date and all other events/dates ($\hat{E} = E \cup E_d$), following its constraint logic (Figure 4.4). Empty outputs (i.e., cases where it cannot deduce a relation) from STAGE are ignored. These relations are introduced as ILP constraints in two ways: (i) adding hard constraints, and (ii) adding soft constraints. Adding hard constraints is done by incorporating the following new constraints into the ILP:

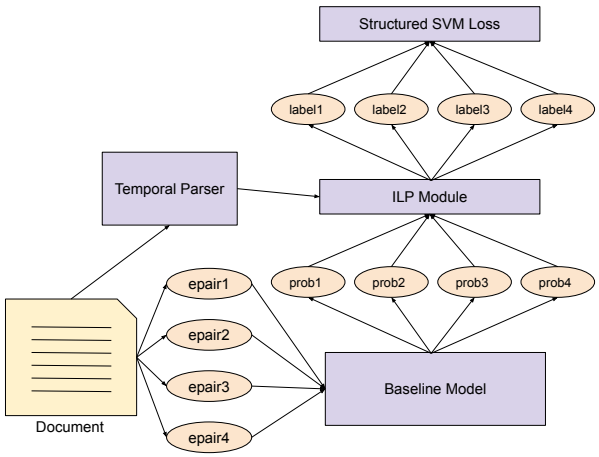


Figure 4.5: Integrated STAGE and BiLSTM+ILP model pipeline.

$$\text{if } TP(e_i, e_j) = r,\ y_{<r,i,j>} = 1 \tag{4.6}$$

$\forall e_i \in E_d, \forall e_j \in \hat{E}, \forall r \in R$. Note that $TP(e_i, e_j) = r$ indicates that the temporal parser predicts that e_i and e_j have the relation r. Adding soft constraints is done by editing the ILP objective to add the following term:

$$\begin{aligned} Obj_{new} = Obj_{old} + \alpha \sum_{e_i \in E_d} \sum_{e_j \in \hat{E}} y_{<TP(e_i,e_j),i,j>} \\ + \left(\frac{1-\alpha}{|R|-1} \right) \sum_{e_i \in E_d} \sum_{e_j \in \hat{E}} \sum_{r \in \hat{R}} y_{<r,i,j>} \end{aligned} \tag{4.7}$$

Here $\hat{R} = R - TP(e_i, e_j)$, which is the set of all relations except for the one predicted by STAGE for pair (e_i, e_j). α is a constant which indicates how much weight we give to the STAGE's prediction. We set it to 0.9 in our experiments because STAGE is a high-precision system as observed during its evaluation on temporal expression extraction (§4.6.3).

4.6.5 Training with TDD-Auto: A Pseudo-Labeling Adaptation Method

In addition to the joint BiLSTM+ILP model, which is a model-centric method, we evaluate the performance of a data-centric adaptation method, to contrast the performance of the two most popular coarse categories of adaptation methods. Since the entire TDD-Auto subset of TDDiscourse is constructed automatically using our temporal inference algorithm, this subset can be considered

Model	TDD-Auto			TDD-Man		
	P	**R**	**F1**	**P**	**R**	**F1**
ZS	15.2	14.4	14.8	4.5	4.3	4.4
BiLSTM+ILP	18.5	17.5	18.0	7.7	7.4	7.6
BiLSTM+ILP+HARD	17.8	16.9	17.3	7.7	7.4	7.6
BiLSTM+ILP+SOFT	19.2	18.2	18.7	7.4	7.1	7.3
BiLSTM+PL	**43.6**	**41.2**	**42.3**	15.8	15.2	15.5
BiLSTM+ILP+PL	43.1	40.7	41.9	**18.1**	**17.4**	**17.8**
BiLSTM+ILP+HARD+PL	42.8	40.5	41.6	15.9	15.3	15.6
BiLSTM+ILP+SOFT+PL	41.5	39.3	40.4	15.5	14.9	15.2
BiLSTM-Sup	48.6	45.9	47.2	28.9	27.7	28.3

Table 4.17: Performance of a baseline temporal ordering model and all adaptation methods on TDDiscourse, when adapting from local event pairs to long-distance event pairs. ZS refers to a zero-shot BiLSTM baseline, which is trained on TimeBank-Dense and tested on TDDiscourse with no adaptation, while BiLSTM-Sup refers to a fully supervised model trained and tested on TDDiscourse.

to have been generated in a distantly supervised manner. Therefore, we conduct experiments in which we incorporate TDD-Auto data during model training, which is equivalent to using distant supervision, a strategy that falls into the pseudo-labeling fine category (PL) of our adaptation method taxonomy.

4.6.6 Results

Table 4.17 shows the performance of the chosen baseline temporal ordering model (BiLSTM) and all adaptation methods on both subsets of TDDiscourse, when adapting from local event pairs (TimeBank-Dense) to long-distance event pairs. ZS indicates the performance of the baseline BiLSTM in a zero-shot setting, in which it is trained on TimeBank-Dense and applied to TDDiscourse with no adaptation modifications. Conversely, BiLSTM-Sup presents the performance of a fully supervised model trained on TDDiscourse, and represents the performance ceiling for the BiLSTM architecture on TDDiscourse. Note that we evaluate three instantiations of the joint BiLSTM+ILP architecture: (i) adding transitivity constraints (BiLSTM+ILP), (ii) adding transitivity and STAGE information as hard constraints (BiLSTM+ILP+HARD), and (iii) adding transitivity and STAGE information as soft constraints (BiLSTM+ILP+SOFT). Moreover, in addition to evaluating adaptation methods in isolation, we also evaluate the performance of combining the pseudo-labeling method (BiLSTM+PL) with all instantiations of the joint BiLSTM+ILP architecture (rows 6-9).

Comparing adaptation methods: From Table 4.17, we can see that both adaptation methods provide performance boosts over a zero-shot baseline. These boosts are particularly pronounced on the TDD-Man subset, on which adaptation method performance is nearly 2-4x the performance

of the zero-shot baseline. Of the adaptation methods tested, pseudo-labeling provides the largest performance increases across both TDD-Auto and TDD-Man (∼3-4x). This performance difference can partly be attributed to the fact that the existing BiLSTM+ILP framework is designed to handle only specific categories of linguistic phenomena that may occur in long-distance pairs. Referring back to the categories uncovered during phenomenon annotation for TDDiscourse (Table 4.11), enforcing consistency using the current ILP formulation is designed to specifically tackle chain reasoning (CR). Chain reasoning is a highly frequent phenomenon in TDD-Man, and fairly frequent in TDD-Auto, which leads to performance improvements on both subsets using the BiLSTM+ILP model. However, pseudo-labeling provides the model access to the entire TDD-Auto training set, which contains the entire range of linguistic phenomena present in long-distance pairs. Therefore, we see much larger performance gains on both subsets using this adaptation method.

Combining adaptation methods: In addition to comparing both adaptation methods, we run experiments combining the pseudo-labeling method with all BiLSTM+ILP instantiations. From Table 4.17, we observe that combining both methods leads to performance improvement on TDD-Man, but minor performance drops on TDD-Auto. This is an interesting result because by adding TDD-Auto training data, the pseudo-labeling method is essentially providing access to "in-distribution" data when the test subset is TDD-Auto, and we see performance degradation from additionally introducing a model-centric method in this scenario. However, when testing on TDD-Man for which TDD-Auto training data is still out of distribution, we see benefits from bringing in a model-centric method. This leaves open a question for further exploration: does adding a model-centric method (to a data-centric method) provide any benefits, if the proportion of available in-distribution data is reduced?

Utility of STAGE constraints: Evaluating three different instantiations of BiLSTM+ILP also helps us identify which heuristic information provides most utility when adapting models trained on local event pairs to long-distance pairs. From Table 4.17, we can see that transitivity alone provides most of the performance benefit, with additional STAGE-generated constraints having little to no effect. A potential explanation for this observation could be that event pairs for which STAGE generates accurate constraints (i.e., finds valuable explicit time cues) are ones that model already performs well on, leaving little scope for performance boosts from adding STAGE constraints. We leave a deeper investigation of this observation up to future work.

Reaching supervised model performance: Finally, Table 4.17 also indicates that despite improved performance, all methods lag behind a fully-supervised BiLSTM. This is an interesting observation for the TDD-Auto subset because the pseudo-labeling method provides access to the TDD-Auto training dataset and therefore should have the capacity to reach fully-supervised performance. The performance lag clearly indicates the existence of conflicting instances in the TimeBank-Dense and TDD-Auto training sets. In prior chapters, instance weighting methods from the hybrid category have been able to identify and filter out such instances. However, all unsupervised instance weighting methods (classifier-based weighting, likelihood-based weighting, etc.) relied largely on lexical similarities between source and target datasets to identify conflicting

instances. This strategy is unlikely to work in the micro-level adaptation scenario, when all macro dimensions are consistent across source and target datasets. This presents an interesting question for future research: how can we learn source-target similarity in an unsupervised manner in a micro-level adaptation scenario, i.e., without resorting to using instances from the target dataset to learn the similarity function?

4.7 Conclusion

In this chapter, we studied micro-level adaptation, i.e. adapting models to handle different linguistic phenomena under the same macro-dimensional scenario. We focused on the task of predicting temporal relations between event pairs in a narrative, targeting long-distance event pairs that have been relegated to the long tail in temporal ordering research and require reasoning about several implicit phenomena such as chain reasoning and maintaining transitivity. As a testbed, we created TDDiscourse, the first dataset focused on discourse-level temporal ordering. To maintain macro dimension consistency, we created TDDiscourse by augmenting the same set of 36 documents used in an existing benchmark (TimeBank-Dense) with long-distance annotations. Our annotation scheme for TDDiscourse handled several issues (e.g., using event coreference) that have not been explicitly addressed in prior work. We also established the challenging nature of this new dataset by benchmarking the performance of 4 state-of-the-art models. All models, on average, performed worse on TDDiscourse than local ordering datasets such as TimeBank-Dense, validating the difficulty of this dataset. Then, we studied the problem of unsupervised micro-level adaptation by training models on local event pairs (TimeBank-Dense) and evaluating their performance on long-distance pairs (TDDiscourse). In addition to a zero-shot baseline, we evaluate adaptation methods from two categories: (i) a pseudo-labeling data-centric method that adds heuristically labeled data from TDD-Auto during training, and (ii) a loss-augmentation model-centric method that incorporates pre-defined heuristics into model loss via ILP constraints. Our results show that both methods improve the performance over a zero-shot BiLSTM baseline, with pseudo-labeling providing higher boosts. Combining both methods demonstrates slightly higher gains than using one method on TDD-Man, but no improvements on TDD-Auto, which can be partly attributed to the restricted phenomenon coverage of our ILP constraints. Lastly, we observe that pseudo-labeling (i.e. training on TimeBank-Dense+TDD-Auto) performs worse on the TDD-Auto test set than training only on TDD-Auto, which indicates that developing non-lexical instance weighting/data selection techniques for micro-level adaptation could be an interesting direction for future research.

CHAPTER 5

Stress Tests: An Evaluation Paradigm for the Long Tail

Despite the field's heavy reliance on benchmarks and leaderboards, our case studies from the previous chapters, as well as the recent body of work on alternate evaluation paradigms (Jia and Liang, 2017; Kaushik et al., 2019; Ribeiro et al., 2020; Gardner et al., 2020) provide evidence that benchmark performance alone is insufficient to provide an accurate picture of model ability to handle various linguistic phenomena. This problem is further exacerbated for the long tail (both macro and micro), since long tail examples are undersampled in traditional benchmarks leading to them being ignored in standard evaluation.

In this chapter, we present stress testing, an evaluation paradigm for the long tail, that can supplement traditional benchmark-based evaluation. Stress testing primarily targets the micro-level long tail since better macro-level long tail evaluation essentially boils down to making sure that benchmarks and leaderboards include a wider range of languages, domains, etc. We define stress tests as supplementary evaluation datasets that test the performance of NLU models on specific sets of micro long tail phenomena required for task reasoning. Constructing such stress tests first requires identification of micro long tail phenomena, for which we propose two strategies: (i) selection via error analysis of state-of-the-art models (Naik et al., 2018), and (ii) selection from human knowledge of the target task (Ravichander et al., 2019). For the task of natural language inference (NLI), we demonstrate via two case studies how these strategies can be used to identify key micro long tail phenomena. We then construct appropriate stress tests, and evaluate state-of-the-art models on these constructed tests, surfacing model weaknesses that benchmark evaluation alone did not highlight. These case studies suggest that supplementing benchmark-based evaluation with alternate paradigms such as stress testing provides a more stringent and insightful evaluation,

especially for the long tail.

5.1 Introduction

Progress in natural language understanding has long been evaluated with the help of standard benchmark datasets, which provide a uniform testbed to compare new modeling developments. In recent years, with the advent of crowdsourcing (Sabou et al., 2014), large-scale standard benchmarks have been created for several core NLU tasks such as question answering (Rajpurkar et al., 2016, 2018), commonsense reasoning (Talmor et al., 2019), natural language inference (Bowman et al., 2015; Williams et al., 2018) and dialog state tracking (Budzianowski et al., 2018). In general, NLU datasets measure model performance on an identically distributed evaluation set, leading to two dominant evaluation paradigms in supervised learning for NLU: (i) Independent and Identically Distributed (IID) IID evaluation paradigm, and (ii) Pretraining-Agnostic and Identically Distributed (PAID) evaluation paradigm.

5.1.1 IID Evaluation Paradigm

Independent and Identically Distributed (IID) evaluation is the most common paradigm in NLU evaluation, as well as supervised natural language processing and machine learning. A central assumption in this paradigm is that the training set used to help a model learn how to perform a task and the test set used to evaluate model performance on the task are *identically distributed*. In practice, this is implemented by collecting a single dataset and randomly splitting it into training, validation and held-out test portions, which ensures that the test split is sourced from the same distribution as the training split.

5.1.2 PAID Evaluation Paradigm

Pretraining-Agnostic and Identically Distributed (PAID) evaluation is a relatively recent paradigm, which is steadily gaining prominence in NLU. As described by Linzen (2020), the PAID evaluation paradigm has surfaced with advancements in learning contextualized word embeddings using deep transformer-based language models (Peters et al., 2018; Devlin et al., 2019). The PAID paradigm consists of three stages:

1. Pretraining a contextualized embedding model using language modeling objectives such as word prediction or next sentence prediction on an arbitrary corpus.
2. Finetuning the model on a task of interest using a training dataset representing the task.
3. Evaluating the model on an identically distributed test dataset, drawn from the same distribution as the training set.

In practice, this is implemented by collecting a single finetuning dataset and randomly splitting it into training, validation and held-out test portions, ensuring that train and test splits are identically distributed. The pretraining corpus however may or may not be sourced from the same distribution as the finetuning dataset.

5.1.3 Drawbacks of Identically Distributed Testing

While reasonable from a machine learning perspective, relying on identically distributed testing on standard benchmarks has some major drawbacks. Benchmark dataset collection procedures, especially those heavily reliant on crowdsourcing, lead to systematic gaps in collected data due to sampling biases and annotator biases, as discussed in Chapter 2. For example, Geva et al. (2019) show that relying on a small pool of crowdworkers to generate a large number of examples negatively affects data diversity. Annotators consistently use a fixed set of language patterns that correlate with the labels, resulting in datasets that are more representative of these patterns than the actual task. Moreover, if datasets are constructed via random sampling, phenomena that naturally occur less frequently will be underrepresented. In the presence of such biases, identically distributed testing leads to the following issues:

- High model performance on the test set does not necessarily signal high performance on an NLP task, because datasets offer limited coverage of the full range of possible macro-level dimensions such as languages, domains, and settings.
- High model performance on the test set does not signal model ability to handle *all* micro-level linguistic phenomena present in the dataset. Datasets may contain higher proportions of certain frequent phenomena, as well as spurious correlations (like the ones introduced by annotator bias). Models can learn to do well on frequent phenomena while sidelining micro long tail phenomena, or leverage high-frequency correlations to do well on the test set, without really understanding the requisite phenomena.

Both issues lead to a critical problem with existing benchmark evaluation - *it overestimates model ability to handle a task or to handle complex linguistic phenomena*. The evaluation problem is further exacerbated for examples that fall into the long tail (both macro and micro), since they are already undersampled in traditional benchmarks, which leads to them being ignored or masked in standard evaluation. To tackle this, we propose that evaluation on standard benchmark datasets should be supplemented with evaluation on additional non-identically distributed evaluation sets, which are systematically constructed to focus on long tail examples. We call these supplementary non-ID evaluation sets as **stress tests**. Each stress test focuses on evaluating model performance on a small subset of micro long tail linguistic phenomena, since macro-level stress testing essentially boils down to broadening benchmarks and leaderboards to include a wider range of languages, domains, etc. This approach allows for fine-grained linguistic phenomenon-driven testing of models

and helps us identify "failure cases", i.e., micro long tail phenomena that contemporary models are unable to handle.

Stress tests are designed using a two stage pipeline: (i) phenomenon selection, and (ii) test construction. The first stage requires identification of micro long tail phenomena to evaluate model performance on, for which we propose two strategies: (i) selection via error analysis of state-of-the-art models, and (ii) selection from human knowledge of the target task. The second stage requires constructing test sets, comprising of naturally occurring or synthetically generated examples, which focus on these selected phenomena. We perform two case studies to show the effectiveness of using supplementary stress test-based evaluation to get a more realistic, detailed overview of model capabilities. The first case study (Naik et al., 2018) focuses on building a stress test-based evaluation for natural language inference, a benchmark task in natural language understanding and sentence representation learning. In this study, we perform phenomenon selection using the error analysis strategy. Our results from this study show that state-of-the-art sentence encoder models which achieve high scores on standard benchmarks are unable to handle several key but low-frequency linguistic phenomena such as antonymy and numerical reasoning. In our second case study (Ravichander et al., 2019), we build a stress test-based evaluation for quantitative reasoning, using an NLI-style format. In this study, phenomenon selection is done based on human knowledge of skills required to perform quantitative reasoning. This evaluation allows us to study how well state-of-the-art NLI models, and a shallow symbolic reasoning baseline, are able to handle specific quantitative reasoning phenomena such as basic arithmetic and quantifiers, which are again uncommon in NLI benchmarks. Our results from this study show that while NLI models and symbolic reasoners fare reasonably well at lexical and numerical aspects of quantitative phenomena respectively, no models possess the ability to successfully combine both skills. Ultimately both case studies demonstrate that stress tests help us to critically evaluate performance of current state-of-the-art models on micro long tail phenomena, reveal clear modeling gaps and unveil areas for further exploration to push the reasoning abilities of NLU models.

5.2 Stress Tests

As defined by Naik et al. (2018), stress tests are supplementary evaluation datasets which test the performance of NLU models on specific sets of micro long tail linguistic phenomena required for task reasoning. Our primary methodological inspiration stems from the work of Jia and Liang (2017), the BIBINLP (Build It Break It, The Language Edition) shared task (Ettinger et al., 2017), and other concurrent work on adversarial evaluation. The main aim of adversarial evaluation is to construct examples that can *attack* state-of-the-art models, i.e., reveal biases and spurious correlations that models rely on, but are not actually helpful for the task. Exact techniques used to construct adversarial evaluation sets differ based on the task of interest and the models being tested.

Our proposed evaluation methodology, while sharing some similar aims, primarily differs from adversarial evaluation in that we stratify test examples into separate sets based on subsets of micro

long tail linguistic phenomena being tested, which infuses systematicity into our evaluation process. These stratified phenomenon-focused test sets are called stress tests, and aim to test systems beyond normal operational capacity on linguistic phenomena in order to identify weaknesses and to confirm that intended specifications are being met. Conceptually, stress tests can also be considered analogous to unit tests in software engineering, since they serve a similar purpose - evaluating whether an NLP model meets specifications in terms of performance on a linguistic phenomenon of interest (Hartman and Owens, 1967; Tretmans, 1999; Beizer, 2003; Pressman, 2005; Nelson, 2009). Like work on adversarial evaluation, this methodology also has the added benefit of being able to isolate high-frequency biases and spurious correlations being leveraged by current NLU models, as we demonstrate through our case studies.

5.2.1 Requirements for Stress Tests

Based on our definition, stress tests should meet the following requirements:

- Preferably, stress tests should be test-only datasets, and not contain training data that can be used to finetune the model. This requirement preserves the non-identically distributed nature of the evaluation sets, and is also echoed by Linzen (2020).
- Each stress test should focus on a single semantic phenomenon, or a restricted subset of related phenomena from the micro-level long tail. This requirement ensures that it is possible to isolate performance on specific micro long tail phenomena of interest.
- Stress test construction procedures should be developed in such a way that the need for crowd-sourced annotation and model-reliant filtering is minimized. This reduces the possibility of annotator biases and model idiosyncrasies seeping into constructed tests.

Note that we intend stress test-based evaluation to supplement, rather than replace, traditional benchmark-based evaluation.

5.2.2 Typical Stress Test Construction Pipeline

Typically, stress test construction follows a two stage pipeline:

1. **Phenomenon Selection:** In this stage, we first choose a set of micro long tail linguistic phenomena to evaluate model performance on. This choice depends on the task of interest and the model capabilities that we are interested in exercising. We propose two strategies for systematic phenomenon selection in this chapter: (i) phenomenon selection by error analysis, as performed in case study I (Naik et al., 2018), and (ii) phenomenon selection from task knowledge, as performed in case study II (Ravichander et al., 2019).

2. **Test Construction:** Post the selection stage, we construct stress tests for each selected phenomenon. Construction procedures for every stress test differ based on the phenomenon being tested and the structure of the task. However, our construction strategies can be broadly categorized into two types: (i) automatic construction using heuristic rules and external knowledge sources, and (ii) expert (or expert-validated) dataset annotation.

The following sections discuss how we follow this two-stage pipeline to create stress tests for natural language inference and numerical reasoning in NLI that meet our requirements, and our observations from evaluating state-of-the-art models on these tests.

5.3 Case Study I: Natural Language Inference

5.3.1 Background: Natural Language Inference

Natural language inference (NLI), also known as recognizing textual entailment (RTE), is the task of determining a directional relationship between two text fragments. The two fragments, usually sentences, are called premise and hypothesis, and the relationship between them captures whether the hypothesis is true, given the premise. The dominant paradigm is to formulate NLI as a 3-way classification task between three labels: (i) *entailment* (hypothesis is true given premise), (ii) *contradiction* (hypothesis is false given premise), and (iii) *neutral* (hypothesis truth value cannot be determined). Recent work has proposed a shift from such categorical labels to subjective probability assessments (Chen et al., 2019), but most benchmark datasets use the 3-way classification paradigm.

NLI has long been considered a benchmark task for natural language understanding research (Cooper et al., 1996; Dagan et al., 2006; Giampiccolo et al., 2007; Dagan et al., 2013; Bowman et al., 2015; Nangia et al., 2017a) since models must learn to reason about several difficult linguistic phenomena, such as scope, coreference, quantification, lexical ambiguity, modality and belief, to perform well at this task (Bowman et al., 2015; Williams et al., 2018). It also serves as an excellent test bed for research on sentence representation learning, since task performance depends heavily on premise and hypothesis representations. Due to its importance, prior work has heavily focused on building datasets for this task (Dagan et al., 2006, 2009, 2013; Marelli et al., 2014a; Bowman et al., 2015; Williams et al., 2018; Conneau et al., 2018; Khot et al., 2018; Romanov and Shivade, 2018). Recently, crowdsourcing has been leveraged to create large-scale benchmark datasets for this task such as the Stanford NLI (SNLI; Bowman et al. (2015)), and Multi-genre NLI (MultiNLI; Williams et al. (2018)) datasets.

Crowdsourced creation of benchmark NLI datasets follows a general procedure. First, a large number of premise sentences are sampled from a corpus of available texts. In the second step, crowdworkers are given a premise sentence and asked to generate novel hypothesis sentences representing the three categories of entailment relations. In an additional validation step, the created premise-hypothesis pairs are shown to multiple annotators (separate from the original hypothesis

author), who provide additional entailment labels for these sentences. Each sentence pair thus gets five labels - one from the hypothesis author, and four labels from additional annotators. Majority voting between these labels is used to decide on the final gold label for each pair, and models are evaluated via classification accuracy on these gold labels. There are some exceptions to this general procedure (eg: SCITAIL; Khot et al. (2018)), which sample hypothesis sentences "from the wild", instead of having them specifically authored by crowdworkers for the entailment task.

Each step in this construction procedure has the potential to introduce biases in the resulting dataset. The premise sampling step can introduce biases towards domains and languages which are high-resource and easy to access. The hypothesis authoring step can introduce biases towards a select set of linguistic phenomena and superficial ways of expressing them, since crowdworkers are interested in maximizing the number of hypotheses they write and can potentially come up with their own quick heuristics to write entailed, contradictory and neutral sentences. Finally, the validation step can introduce biases towards unambiguous phenomena, since those phenomena are most likely to have agreement between at least three labels. Another interesting source of bias is cases with discrepancies between gold labels and author labels. Since there are four labels from additional annotators, there may be cases where the majority label is not the same as the label provided by the original author. For example, in SNLI and MultiNLI, 6.8% and 5.6% cases respectively show this discrepancy. This has the potential to bias datasets towards surface-level readings of the sentence pair instead of the original author intent.

Despite these potential biases, benchmark NLI datasets have been incorporated into general-purpose language understanding benchmarks like GLUE and SuperGLUE (Wang et al., 2019c,b) and are widely used to evaluate sentence representation learning methods. State-of-the-art deep learning-based sentence encoder models (Nie and Bansal, 2017; Chen et al., 2017b; Conneau et al., 2017; Balazs et al., 2017) have achieved consistently high accuracies on SNLI and MultiNLI, which may lead us to believe that these models excel at performing the NLI task across various genres of text. However, in the presence of potential sampling biases, we need to question what conclusions we can draw regarding model ability to solve the *task*, based on its performance on a *dataset*. This is particularly crucial when train and test sets are identically distributed, because machine learning models are known to exploit idiosyncrasies of the data construction process, allowing NLI models to achieve high accuracy without learning the underlying reasoning involved in the task (Levesque, 2014; Rimell et al., 2009; Papernot et al., 2017). Therefore, we construct and use a stress test-based evaluation to answer this question: does good model performance on NLI benchmark *datasets* reflect competence at various types of reasoning required to do well on the *task*?

To construct stress tests, we first select a set of phenomena to test by examining the errors of the best-performing sentence encoder model on MultiNLI (Nie and Bansal, 2017) to identify phenomena that it finds challenging (§5.3.2). We then automatically construct stress tests for each phenomenon (§5.3.3), making it possible to perform evaluation on a phenomenon-by-phenomenon basis. On benchmarking the performance of four state-of-the-art models on MultiNLI on our constructed stress tests, we observe that models exhibit huge performance drops across stress tests,

especially for phenomena such as antonymy and numerical reasoning (§5.3.4). Our results demonstrate that using stress tests as a supplementary evaluation, in addition to traditional evaluation, can help us identify model weaknesses and strengths on various phenomena in a more fine-grained manner. [1]

5.3.2 Phenomena Selection by Error Analysis

To select a set of micro long tail phenomena to "stress test" NLI models on, we rely on a manual analysis of errors made by the shortcut-stacked sentence encoder model from Nie and Bansal (2017), which was the top-performing model on MultiNLI at the RepEval shared task (Nangia et al., 2017b). The Multi-Genre Natural Language Inference (MultiNLI) corpus is a crowd-sourced collection of 433,000 sentence pairs annotated for textual entailment. This corpus contains sentence pairs from texts drawn from 10 different genres of spoken and written English, and supports a distinctive cross-genre generalization evaluation. Of these 10 genres, only 5 are present in the training set, whereas the development and test sets contain all 10 genres. Models thus can be evaluated on both the *matched* test examples, which are derived from the same sources as those in the training set, and on the *mismatched* examples, which do not closely resemble any seen at training time.

For our error analysis, we sample 100 misclassified examples from both matched and mismatched sets, analyze their potential sources of errors, and develop a typology of common reasons for error. In the end, the reasons for errors can broadly be divided into the following categories:

1. **Word Overlap**: Large lexical overlap between premise and hypothesis sentences causes the model to wrongly predict entailment, even if the sentences are unrelated. On the other hand, very little word overlap causes a prediction of neutral instead of entailment. For example:
    - **Premise:** And, could it not result in a decline in Postal Service volumes across–the–board?
    - **Hypothesis:** There may not be a decline in Postal Service volumes across–the–board.
    - **Prediction Error:** Entailment, instead of neutral
2. **Negation**: The presence of strong negation words ("no", "not"), especially in the hypothesis, causes the model to predict contradiction for neutral or entailed statements. For example:
    - **Premise:** Enthusiasm for Disney's Broadway production of The Lion King dwindles.
    - **Hypothesis:** The broadway production of The Lion King is no longer enthusiastically attended.
    - **Prediction Error:** Contradiction, instead of entailment

[1] Stress tests and other resources available at https://abhilasharavichander.github.io/NLI_StressTest/

3. **Antonymy**: Premise-hypothesis pairs containing antonyms (instead of explicit negation) are not detected as contradiction by the model. For example:

    - **Premise:**‘Have her show it," said Thorn.
    - **Hypothesis:** Thorn told her to hide it.
    - **Prediction Error:** Entailment, instead of contradiction

4. **Numerical Reasoning**: For some premise-hypothesis pairs, the model is unable to perform reasoning involving numbers or quantifiers for correct relation prediction. For example:

    - **Premise:** Deborah Pryce said Ohio Legal Services in Columbus will receive a $200,000 federal grant toward an online legal self-help center.
    - **Hypothesis:** A $900,000 federal grant will be received by Missouri Legal Services, said Deborah Pryce.
    - **Prediction Error:** Entailment, instead of contradiction

5. **Length Mismatch**: The premise is much longer than the hypothesis and this extra information acts as a distraction for the model. For example:

    - **Premise:** So you know well a lot of the stuff you hear coming from South Africa now and from West Africa that’s considered world music because it’s not particularly using certain types of folk styles.
    - **Hypothesis:** They rely too heavily on the types of folk styles.
    - **Prediction Error:** Neutral, instead of contradiction

6. **Grammaticality**: The premise or the hypothesis is ill-formed due to spelling errors or incorrect subject-verb agreement. These minor issues act as distractors for models. For example:

    - **Premise:** So if there are something interesting or something worried, please give me a call at any time.
    - **Hypothesis:** The person is open to take a call anytime.
    - **Prediction Error:** Entailment, instead of neutral

7. **Real-World Knowledge**: These examples are hard to classify without some factual, real-world knowledge. For example:

    - **Premise:** It was still night.
    - **Hypothesis:** The sun hadn’t risen yet, for the moon was shining daringly in the sky.
    - **Prediction Error:** Neutral, instead of entailment

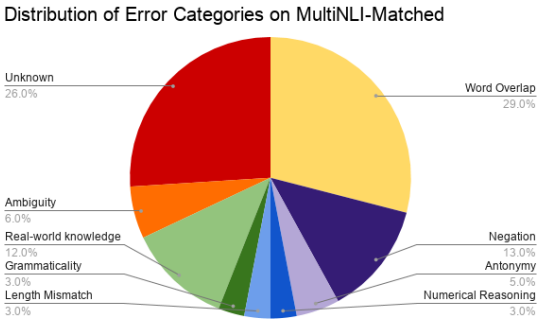


Figure 5.1: Distribution of error categories on MultiNLI-Matched.

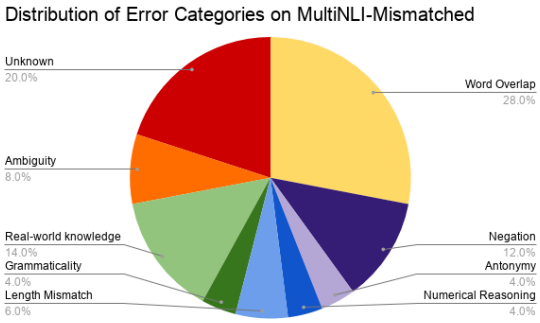


Figure 5.2: Distribution of error categories on MultiNLI-Mismatched.

8. **Ambiguity**: For some instances, the gold label is ambiguous to humans, while the model prediction seems reasonable. These are the most difficult cases. For example:
   - **Premise:** Outside the cathedral you will find a statue of John Knox with Bible in hand.
   - **Hypothesis:** John Knox was someone who read the Bible.
   - **Prediction Error:** Neutral, instead of entailment
9. **Unknown**: No obvious source of error is discernible in these samples. For example:
   - **Premise:** We're going to try something different this morning, said Jon.
   - **Hypothesis:** Jon decided to try a new approach.
   - **Prediction Error:** Contradiction, instead of entailment

Figures 5.1 and 5.2 show the distribution of these error categories in both matched and mismatched sets. Some error categories such as real world knowledge, negation scope, and antonymy, are well-known "hard" linguistic phenomena across several tasks in natural language understanding, and have garnered significant interest in formal semantics literature (Kroch, 1974; Muehleisen, 1997; Murphy, 2003; Moscati, 2006; Brandtler, 2006). These phenomena have long been suspected to be challenging for entailment models (Jijkoun and De Rijke, 2006; LoBue and Yates, 2011; Roy, 2017). Other error categories such as word overlap and length mismatch are "artifacts" of recent crowdsourcing strategies used for dataset construction, and have been identified as major distractors for NLI models by other concurrent work (Gururangan et al., 2018; Poliak et al., 2018; McCoy et al., 2019).

5.3.3 Constructing Stress Tests

Based on our typology of error categories, we develop methodologies to construct stress tests for phenomena exhibited by categories 1-6. Notably, we focus only on spelling errors within

Premise: I love the Cinderella story.
Hypothesis: I hate the Cinderella story.
Label: Contradiction

Table 5.1: Sample sentence pair from antonymy stress test.

grammaticality. We omit the real world knowledge category as it is not trivial to create a large dataset without human input, the ambiguity category because it is unreasonable to expect models to handle such cases, and the unknown category because it does not correspond to a particular phenomenon. We organize stress tests for categories 1-6 into three major classes, based on the type of reasoning required for the model to do well on the test. The first class (*competence tests*) require models to possess the ability to reason about complex semantic phenomena such as quantities and antonymy, which are often low-frequency in standard NLI benchmarks, which pushes them into the micro long tail. The second class (*distraction tests*), requires models to possess the ability to ignore high-frequency artifacts such as lexical similarity or presence of negation words, introduced by the dataset construction process. This class covers the word overlap, negation and length mismatch error categories. The final class (*noise tests*) requires models to be robust to minor perturbations or noise and consists of our spelling error test. For stress test construction, we use three techniques: heuristic rules with external knowledge sources (for competence tests), a propositional logic framework (for distraction tests) and randomized perturbation (for noise tests). The following subsections describe our stress test construction in detail, along with some examples, and Appendix C presents additional examples from all categories.

Competence Test Construction

Antonymy: To construct a test set for antonymy, we consider every sentence from premise-hypothesis pairs in the MultiNLI development set independently. We perform word-sense disambiguation for each adjective and noun in the sentence using the Lesk algorithm (Lesk, 1986). We then randomly sample an antonym for the word from WordNet (Miller, 1995). The original sentence and the sentence with the word substituted by its antonym become a new premise-hypothesis pair in our test set, with the label *contradiction*. Table 5.1 shows an example pair from this construction process. This process results in the construction of 1561 and 1734 premise-hypothesis pairs for matched and mismatched sets respectively.

Substituting a word with its antonym may not always result in a contradiction. For example, this may happen in case of sentences with modalities, belief, conjunction or even conversational text such as *"They can change the tone of people's voice yes."*, *"They can change the tone of people's voice no."*. Coreference issues, word substitution in metaphors or failure of word sense disambiguation might also lead to non-contradictory pairs. Hence, we perform a validation study in which three annotators were provided 100 random samples from the stress test set to evaluate for correctness. At least two annotators agreed on 86% of the labels being contradiction. We also

Premise: Tim has 350 pounds of cement in 100, 50, and 25 pound bags. **Hypothesis:** Tim has less than 750 pounds of cement in 100, 50, and 25 pound bags. **Label:** Entailment
Premise: Tim has 350 pounds of cement in 100, 50, and 25 pound bags. **Hypothesis:** Tim has 750 pounds of cement in 100, 50, and 25 pound bags. **Label:** Contradiction
Premise: Tim has 750 pounds of cement in 100, 50, and 25 pound bags. **Hypothesis:** Tim has 350 pounds of cement in 100, 50, and 25 pound bags. **Label:** Neutral

Table 5.2: Sample sentence pairs from numerical reasoning stress test.

evaluated grammaticality of our constructions, with at least two annotators agreeing on 87% cases being grammatical.

Numerical Reasoning: Creating a stress test for numerical reasoning from MultiNLI is non-trivial as most sentences from the MultiNLI development set do not contain quantities (providing further proof of the long tail nature of quantitative reasoning). Hence, we use a different data source to sample premise sentences: AQuA-RAT, a dataset specifically focused on algebraic word problems along with rationales for their solutions (Ling et al., 2017). Word problems from AQuA-RAT are quite complicated, involving concepts such as probability, geometry and theoretical proofs, which a general-purpose NLI models cannot reasonably be expected to solve. Hence, we perform some preprocessing to filter out such samples and generate a reasonable set of premise sentences.

For preprocessing, we first discard problems which do not have numerical answers or have long rationales (>3 sentences) as such problems are inherently complex. We then split all problems into individual sentences and discard sentences without numbers, resulting in a set of 40,000 sentences. From this set, we discard sentences which do not contain at least one named entity (we consider "PERSON", "LOCATION" and "ORGANIZATION"), since such sentences mostly deal with abstract concepts.[2] This results in a set of 2500 premise sentences. For each premise, we generate entailed, contradictory and neutral hypotheses using heuristic rules:

1. **Entailment:** Randomly choose and change one numerical quantity from the premise, prefixing it with the phrase "less than" or "more than" based on whether the new number is higher or lower.
2. **Contradiction:** Perform one of two actions with equal probability: randomly choose a numerical quantity from the premise and change it, or randomly choose a numerical quantity from the premise and prefix it with "less than/ more than" without changing it.
3. **Neutral:** Flip the corresponding entailed premise-hypothesis pair.

[2]For example, "Find the smallest number of five digits exactly divisible by 22, 33, 66 and 44".

Premise: Possibly no other country has had such a turbulent history. **Hypothesis:** The country's history has been turbulent and true is true. **Label:** Entailment
Premise: Possibly no other country has had such a turbulent history. **Hypothesis:** The country's history has been turbulent and false is not true. **Label:** Entailment
Premise: Possibly no other country has had such a turbulent history and true is true and true is true and true is true and true is true and true is true. **Hypothesis:** The country's history has been turbulent. **Label:** Entailment

Table 5.3: Sample sentence pairs from word overlap, negation and length mismatch distraction tests.

Using these rules, we generate a set of 7,596 premise-hypothesis pairs testing models on their ability to perform numerical reasoning. Table 5.2 shows some sample pairs from this set. We further validate this set by instructing three human annotators to evaluate 100 randomly sampled examples for difficulty, grammaticality and label correctness (since the labels are automatically generated). At least two annotators agreed with our generated label for 91% of the samples. Additionally, at least two annotators agreed on 92% of the examples being grammatical, and 98% being trivial numerical reasoning for humans.

Distraction Test Construction

This class includes stress tests for word overlap, negation and length mismatch, which test model ability to avoid getting distracted by shallow but high-frequency artifacts such as lexical similarity or strong negation words. Models usually learn to exploit such cues to achieve high performance since they have strong but spurious correlations with gold labels, but this reliance on shallow reasoning can end up with high dataset performance at the expense of developing a true understanding of the task, as we demonstrate. We use a framework inspired by propositional logic to construct these stress tests.

Propositional Logic Framework: Assume a premise p and a hypothesis h. For entailment, $(p \Rightarrow h) \implies (p \wedge True \Rightarrow h)$ since $(p \wedge True = p)$. Similarly if p and h are contradictory/neutral, then they remain so even after adding a tautology in conjunction to the premise. In other words, if the premise or hypothesis is in conjunction with a statement that is independently true in all worlds, the entailment relationship is preserved. The next step is to construct such tautological statements which are true in all worlds and then append them to premise or hypothesis sentences to construct distraction tests. We use simple tautologies, which do not contain words that share any topical significance with the premise or hypothesis. Specific details for our sets are as follows:
Word Overlap: For this set, we append the tautology *"and true is true"* to the end of the hypothesis

Premise: As he emerged, Boris remarked, glancing up at teh clock: "You are early".
Hypothesis: Boris had just arrived at the rendezvous when he appeared.
Label: Neutral

Table 5.4: Sample sentence pair from spelling error stress test.

sentence for every example in the MultiNLI development set.

Negation: For this set, we append the tautology *"and false is not true"*, which contains a strong negation word (*"not"*), to the end of the hypothesis sentence for every example in the MultiNLI development set.

Length Mismatch: For this adversarial set, we append the tautology *"and true is true"* five times to the end of the premise sentence for every example in the MultiNLI development set. We modify the premise sentence in this case as we hypothesize that errors in this category mainly arise due to the premise sentence being unwieldy. Table 5.3 shows some examples from these test sets.

A natural concern is that sentence pairs obtained from such constructions are unnatural (Grice, 1975), and could make the NLI task more distracting for humans as well. To study this, we run a human evaluation where three annotators are shown premise-hypothesis pairs from these sets and instructed to label the relation. On word overlap, we find that the provided label has 91% agreement with the gold label. For length mismatch, the provided label has 85% agreement with gold. This is similar to the agreement reported in Williams et al. (2018), leading us to believe the constructed examples are not too unnatural or difficult. The constructions also remain grammatical; after annotating 100 samples from our adversarially generated set, only two were deemed ungrammatical, and both were because of reasons unrelated to our perturbations.

Noise Test Construction

This class consists of a stress test which evaluates model robustness to spelling errors. Spelling errors occur often in MultiNLI data, due to the involvement of Turkers and noisy source text (Ghaeini et al., 2018), which is problematic as many NLI systems rely heavily on word embeddings. Inspired by Belinkov and Bisk (2018), we construct a stress test for spelling errors by performing two types of perturbations on a word sampled randomly from the hypothesis: (i) random swap of adjacent characters within the word (for example, *"I saw Tipper with him at teh movie."*), and (ii) random substitution of a single alphabetical character with the character next to it on the English keyboard (for example, *"Agencies have been further restricted and given less choice in selecting contractimg methods"*. Entailment labels for the perturbed sentence pairs remain unchanged. Table 5.4 shows a sample sentence pair from this set.

5.3.4 Experiments and Analysis

We evaluate the following sentence-encoder models, which achieve strong performance on MultiNLI, on our stress tests:

- **Nie and Bansal (2017) (NB)**: This model uses a sentence encoder consisting of stacked BiLSTM-RNNs with shortcut connections and fine-tuning of embeddings. It achieves the top non-ensemble result in the RepEval-2017 shared task (Nangia et al., 2017b).
- **Chen et al. (2017b) (CH)**: This model also uses a sentence encoder consisting of stacked BiLSTM-RNNs with shortcut connections. Additionally, it makes use of character-composition word embeddings learned via CNNs, intra-sentence gated attention and ensembling to achieve the best overall result in the RepEval-2017 shared task.
- **Balazs et al. (2017) (RiverCorners - RC)**: This model uses a single-layer BiLSTM with mean pooling and intra-sentence attention.
- **Conneau et al. (2017) (InferSent - IS)**: This model uses a single-layer BiLSTM-RNN with max-pooling. It is shown to learn robust universal sentence representations which transfer well across several inference tasks.

Additionally, we also set up two simple baseline models:

- **BiLSTM:** The simple BiLSTM baseline model described by Nangia et al. (2017b).
- **CBOW:** A bag-of-words sentence representation from word embeddings.

Table 5.5 shows the classification accuracy of all six models on our stress tests and the original MultiNLI development set. We see that performance of all models drops across all stress tests, indicating that while models may be doing well on the MultiNLI dataset, there are visible gaps in their ability to tackle crucial phenomena required for the NLI task, especially micro long tail phenomena. On competence stress tests, no model is a clear winner, with **RC** and **CH** performing best on antonymy and numerical reasoning respectively. On distraction tests, **CH** is the best-performing model, suggesting that their gated-attention mechanism can handle shallow lexical distractions to some extent. Interestingly, our **BiLSTM** baseline is the second-best model on two out of three distraction tests. On the noise test, **CH**, **RC** and both baselines **[BiLSTM;CBOW]** do not show much performance degradation, most likely due to the benefit of subword modeling via character-CNNs and the use of mean pooling. We perform further analyses of model performance on each class of tests to obtain more insight into the the kinds of errors they make, and what linguistic phenomena are still hard for most models.

System	Original MultiNLI Dev		Competence Test			Distraction Test						Noise Test	
			Antonymy		Numerical Reasoning	Word Overlap		Negation		Length Mismatch		Spelling Error	
	Mat	Mis	Mat	Mis		Mat	Mis	Mat	Mis	Mat	Mis	Mat	Mis
NB	74.2	74.8	15.1	19.3	21.2	47.2	47.1	39.5	40.0	48.2	47.3	51.1	49.8
CH	73.7	72.8	11.6	9.3	30.3	58.3	58.4	52.4	52.2	63.7	65.0	68.3	69.1
RC	71.3	71.6	36.4	32.8	30.2	53.7	54.4	49.5	50.4	48.6	49.6	66.6	67.0
IS	70.3	70.6	14.4	10.2	28.8	50.0	50.2	46.8	46.6	58.7	59.4	58.3	59.4
BiLSTM	70.2	70.8	13.2	9.8	31.3	57.0	58.5	51.4	51.9	49.7	51.2	65.0	65.1
CBOW	63.5	64.2	6.3	3.6	30.3	53.6	55.6	43.7	44.2	48.0	49.3	60.3	60.6

Table 5.5: Classification accuracy (%) of state-of-the-art models on our constructed stress tests. Accuracies shown on both matched and mismatched categories for each stress set developed from MultiNLI. For reference, random baseline accuracy is 33%.

What insights can we obtain from model performance on competence tests?

Model Performance on Antonymy: Table 5.5 shows that all models perform poorly on antonymy. **RC** achieves the best performance, with 36.4% and 32.8% on matched and mismatched sets respectively which is slightly higher than random performance. However, none of the other models beat a random baseline. Digging deeper into model errors, we observe that high amount of word overlap in this test causes models to overpredict entailment, accounting for, on average, 86.4% and 87.6% of total errors on matched and mismatched sets respectively. We present the exact proportion of false entailment and false neutral errors in Table 5.6. Keep in mind there is only one gold class in this category: contradiction. As expected, all four models make a high amount of false entailment errors because they notice high amounts of lexical similarity between the premise and the hypothesis.

System	C-E Errors		C-N Errors	
	Mat	Mis	Mat	Mis
NB	79.83	82.40	20.17	17.60
CH	99.78	99.75	0.22	0.25
RC	66.67	68.50	33.33	31.50
IS	99.40	99.81	0.60	0.19

Table 5.6: Percentage of C-E and C-N errors on antonymy test.

We further study which antonym pairs are easy and difficult for models by examining the errors of the best and worst performing models on this test **[RC;CH]**. On 982 samples where both models fail, we find 617 unique antonym pairs, and on 171 samples where both models succeed, we find 84 unique antonym pairs. 89.8% of the "easy" and 57.2% of the "hard" antonym pairs appear in a contradiction relation within the training data, suggesting that models succeed on easy antonym-pairs seen in the training data but struggle to generalize. In addition to frequency, we

study error variation by antonym type. We randomly sample 100 examples where both models fail and 100 samples where both succeed, and manually annotate whether the antonym present was gradable, relational or complementary. Among successful examples, 99% are complementary antonyms with only one relational antonym. Amongst the failure cases, 20% are relational antonym pairs, 73% are complementary and 7% are gradable, suggesting that models find relational and gradable antonyms hard, but get complementary antonyms both right and wrong. Finally, we examine differences between models by analyzing examples classified correctly by the best model which are not handled by the worst. We find that antonym pairs recognized by the weaker model occur, on average, nearly twice as often in the training data as antonym pairs recognized by the stronger model, suggesting that **RC** is able to learn antonymy from fewer examples (though these examples must be present in training data).

Model Performance on Numerical Reasoning: Table 5.5 shows that all models exhibit a significant performance drop on numerical reasoning, with none achieving an accuracy better than random (33%). We analyze the predictions of the best and worst performing models on this test **[BiLSTM;NB]**. The biggest source of common errors for both models (1703 out of 4337 errors) is incorrectly classifying neutral pairs as entailment, which arises because our construction technique flips entailed premise-hypothesis pairs to create neutral pairs, leading to high word overlap for neutral pairs. Our constructions also lead to high word overlap for contradiction pairs, leading to a large number of C-E errors for both models (1695 out of 4337 errors). Thus, 78.3% of all errors are caused due to the models falsely predicting entailment. Most of the remaining errors are caused by entailment examples containing the phrases "more than" or "less than" being incorrectly classified as contradiction. We speculate that this behavior could arise as these phrases are often used by crowdworkers to create contradictory examples in the original MultiNLI data, fooling models into marking examples with this phrase as "contradiction" without reasoning about involved quantities. Our observations suggest that models do not perform quantitative reasoning, but simply rely on word overlap and other shallow lexical cues for prediction. We explore this in more detail in our second case study on numerical reasoning in NLI, which includes tests that evaluate model ability to perform and reason about simple mathematical operations such as addition and subtraction.

What insights can we obtain from model performance on distraction tests?

The design of our distraction tests allows us to evaluate model robustness to: (i) decreasing lexical similarity between premise-hypothesis pairs, and (ii) presence of strong negation words in sentence pairs.

Effect of Decreasing Lexical Similarity: Due to our construction methodology (i.e., appending tautologies), accuracy on the word overlap and length mismatch tests demonstrates the effect of decreasing lexical similarity on model performance. Table 5.5 shows accuracy drops for all models

System	MultiNLI Dev		Word Overlap		Length Mismatch	
	Mat	Mis	Mat	Mis	Mat	Mis
NB	33.2	33.1	43.2	38.3	46.0	46.9
CH	32.9	31.7	84.7	85.3	65.8	65.8
RC	37.1	39.1	74.3	83.3	74.2	79.5
IS	34.7	31.4	86.3	87.0	43.5	44.2
BiLSTM	38.5	37.9	83.2	81.9	75.9	79.1
CBOW	33.9	30.2	74.5	72.3	54.7	59.9

Table 5.7: % of FALSE NEUTRAL cases among total errors on MultiNLI development set, word overlap test and length mismatch test.

on both tests. This drop is lower for **CH**, suggesting that their gated attention mechanism might help in focusing on relevant parts of the sentence. The significant decrease in accuracy on these tests indicates that NLI models use lexical similarity as a strong signal for entailment prediction, failing which models default to predicting neutral. To provide further justification, we compare the proportion of false neutral errors for all models on word overlap and length mismatch stress sets vs. the original MultiNLI development set. As shown in Table 5.7, we find that it increases for all models on both sets.

Effect of Introducing Strong Negation Words: Table 5.5 shows results on the negation test, and we see that all state-of-the-art models perform poorly, with accuracies decreasing by 23.4% and 23.38%, on average, on matched and mismatched sets respectively. However, comparing the number of E-C (entailment predicted as contradiction) and N-C (neutral predicted as contradiction) errors for these models on the negation test vs. the original MultiNLI development set, we do not find an increase in these error types on negation. Instead, we observe an increase in false neutral errors for all models. We hypothesize that this could occur due to the introduction of extra words ("false", "is" and "true") apart from "not", indicating that decreasing lexical similarity has a stronger effect on models than introducing negation.

Training with Distraction: Finally, we perform an additional experiment to study whether models can learn to ignore shallow lexical distractions if they are trained on distracting examples, which could be a simple strategy to improve robustness. To do so, we generate an equivalent sample containing the negation distraction ("false is not true") for every sample in the training data, and retrain **NB** and **BiLSTM** on the union of these examples and original training data. We evaluate the performance of the retrained models on three tests: the original MultiNLI development set, the negation stress test and a new distraction test created using a different negation tautology *"green is not red"* (DIFF TAUT). We observe that **NB** shows performance degradation across all tests, but training **BiLSTM** on distraction data helps it become robust to the tautology it was trained on. However, it collapses when evaluated on a different tautology. This shows that when trained on

System	BiLSTM		NB	
	Mat	Mis	Mat	Mis
MultiNLI Dev	70.2	70.4	66.6	66.6
NEGATION	68.9	70.4	49.3	48.7
DIFF TAUT	49.0	49.3	49.9	49.7

Table 5.8: Effect of training on distraction data on original DEV set, original distraction set and new distraction set.

distractions, models simply learn to ignore specific distracting phrases, instead of learning to ignore distracting patterns. However, ignoring such distraction patterns is something humans do naturally. Models should not have to train on specific distraction phrases to succeed on this evaluation.

What insights can we obtain from model performance on noise tests?

Our noise test results in Table 5.5 show that **NB** and **IS** exhibit a huge decrease in accuracy, since both models rely on word embeddings. Other models show little performance degradation on this test. **CH** performs subword modeling via character-level CNNs, which provides robustness towards perturbation attacks. **RC** and **BiLSTM** perform well despite relying on word embeddings since both use mean pooling, which might reduce the effect of single-word edits on the final representation. **CBOW** is also very robust to this test, which can arise from the fact that it sums word embeddings to create the final sentence embedding, diluting the effect of changing a single word on final model performance.

We further analyze the performance of all four sentence encoder models under various perturbation settings for noise introduction. In addition to exploring two types of perturbations (ADJSWAP and KBSWAP as described earlier), we perform perturbations on only function words (conjunctions, pronouns and articles), and on only content words (nouns and adjectives) in the hypothesis to study the effects. We do not address perturbations in verbs and adverbs in the content word vs. function word analysis. The results from these experiments are presented in Table 5.9.

System	ADJSWAP		KB SWAP		CN SWAP		FN SWAP	
	Mat	Mis	Mat	Mis	Mat	Mis	Mat	Mis
NB	43.0	42.9	47.7	47.9	51.1	49.8	49.7	49.6
CH	68.24	68.1	68.5	68.3	68.3	69.1	69.9	70.3
RC	66.6	66.4	67.0	66.8	66.6	67.0	68.4	68.4
IS	57.8	58.6	57.7	58.7	58.3	59.4	57.5	57.6

Table 5.9: Model performance on different perturbation techniques for noise introduction.

We observe that there is no significant difference between perturbing a function word or a content word, which is surprising. One hypothesis is that content words can often be named entities

for which the models already do not find word embeddings. We also do not find a considerable difference in performance between the different kinds of perturbations but this is expected behaviour as most models use word embeddings, and irrespective of the type of perturbation, these will just be categorized as unknown words.

Final observations from case study I: Our analyses of performance of state-of-the-art sentence encoder models on a benchmark NLI dataset (MultiNLI) and our proposed stress tests leave us with the following key observations:

- Performance on a benchmark dataset does not always provide a complete picture of model ability to handle all requisite phenomena need for the actual task being tested. This is particularly true when the benchmark dataset construction process is prone to sampling and annotator biases.
- Our proposed evaluation methodology of using stress tests helps us isolate important micro long tail phenomena that models do not capture, as well as identify high-frequency spurious artifacts that act as distractors for models. They also allow us to perform comparative analyses of model architectures and obtain actionable insights into which long tail phenomena are still hard for most state-of-the-art models.
- Standard benchmark evaluation in conjunction with a non-identically distributed evaluation methodology such as stress tests provides a more stringent and insightful evaluation process that is harder for machine learning models to fool.

5.4 Case Study II: Numerical Reasoning in NLI

5.4.1 Background: Numerical Reasoning

Numerical reasoning, or quantitative reasoning, is a higher-order reasoning skill that an intelligent natural language understanding system can reasonably be expected to handle. Humans reason with numbers in many day-to-day tasks ranging from handling currency to reading news articles to understanding sports results, elections and stock markets. Numbers are used to communicate information accurately, and so learning to reason with them is an essential competence in understanding natural language (Levinson, 2001; Frank et al., 2008; Dehaene, 2011). In the field of NLU, numerical reasoning has typically been studied via the task of solving arithmetic word problems (Hosseini et al., 2014; Mitra and Baral, 2016; Zhou et al., 2015; Upadhyay et al., 2016; Huang et al., 2017; Kushman et al., 2014; Koncel-Kedziorski et al., 2015; Roy and Roth, 2015; Roy, 2017; Ling et al., 2017). An important limitation of this task is that word problems primarily focus on testing model ability to perform arithmetic reasoning, while the requirement for linguistic reasoning and factual world knowledge is limited as the text is concise, straightforward, and self-contained (Hosseini et al., 2014; Kushman et al., 2014). However, NLU systems capable of performing numerical

reasoning must be able to handle the full complexity of language and tackle the intricate interplay between everyday language and numbers beyond arithmetic reasoning, which includes phenomena such as approximation, ordinality, scalar implicature, etc. Motivated by these requirements, we study the numerical reasoning skill through the lens of the natural language inference task.

As described previously, natural language inference (NLI), or recognizing textual entailment (RTE) (Cooper et al., 1996; Condoravdi et al., 2003; Bos and Markert, 2005; Dagan et al., 2006), is a benchmark task in natural language understanding, wherein a model determines if a natural language hypothesis can be justifiably inferred from a given premise. This is most commonly posed as a three-way classification decision where the hypothesis can be inferred to be true (entailment), false (contradiction) or cannot be determined (neutral). Making such inferences often involves reasoning about quantities. Consider the following example:
Premise: With 99.6% of precincts counted , Dewhurst held 48% of the vote to 30% for Cruz.
Hypothesis: Lt. Gov. David Dewhurst fails to get 50% of primary vote.
To conclude that the hypothesis is true given the premise, a model must reason that since 99.6% precincts are counted, even if all remaining precincts were to vote for Dewhurst, he would fail to get 50% of the primary vote. Such examples requiring inferences involving quantities frequently crop up in existing NLI datasets. de Marneffe et al. (2008) find that in a corpus of real-life contradiction pairs collected from Wikipedia and Google News, *29%* contradictions arise from numeric discrepancies, which requires reasoning about quantities to detect them accurately. In the RTE-3 (Recognizing Textual Entailment) development set, numeric contradictions make up 8.8% of contradictory pairs. Naik et al. (2018) find that model inability to do numerical reasoning causes 4% of errors made by state-of-the-art models. Moreover, prior work arguing for a systematic knowledge-oriented approach to solving NLI by evaluating specific semantic analysis tasks, has identified quantitative reasoning as one of the focus areas (Sammons et al., 2010; Clark, 2018).

There is no scarcity of large-scale benchmark NLI datasets, which do not focus on a specific skill (Bowman et al., 2015; Williams et al., 2018; Khot et al., 2018; Conneau et al., 2018; Romanov and Shivade, 2018), especially since NLI has attracted community-wide interest as a stringent test for natural language understanding (Cooper et al., 1996; Fyodorov; Glickman et al., 2005; Haghighi et al., 2005; Harabagiu and Hickl, 2006; Romano et al., 2006; Dagan et al., 2006; Zanzotto et al., 2006; Giampiccolo et al., 2007; Malakasiotis and Androutsopoulos, 2007; MacCartney, 2009; de Marneffe et al., 2009; Dagan et al., 2010; Angeli and Manning, 2014; Marelli et al., 2014b). But standard identically distributed evaluation on benchmark NLI datasets does not provide an accurate picture of model ability to handle a particular skill such as quantitative reasoning. In particular, the hypothesis authoring step used to obtain entailed, neutral and contradictory hypothesis sentences from crowdworkers encourages biases towards shallow expressions of linguistic phenomena. For example, Gururangan et al. (2018) show that when provided premise sentences containing numbers, a common strategy followed by crowdworkers is to replace exact numbers with approximates. More complex quantitative phenomena such as scalar implicature and arithmetic reasoning are rarely instantiated. Recognizing this flaw in standard benchmark dataset evaluation has led to

the development of supplementary challenge test sets for specific linguistic phenomena such as lexical inference with hypernymy, co-hyponymy, antonymy (Glockner et al., 2018; Naik et al., 2018). Despite this, there has been limited work on building evaluation sets specifically focused on numerical reasoning in NLI. Bentivogli et al. (2010) create several specialized phenomenon-specific NLI datasets, but feature only 6 examples with quantitative reasoning. Roy (2017) propose a dataset and model for a related sub-task called *quantity entailment*, which aims to determine if a single given quantity can be inferred from a sentence, instead of trying to infer the relationship between two sentences. Naik et al. (2018) build a stress test for numerical reasoning, but this test only covers quantifiers, leaving out many other quantitative phenomena.

To address this gap, we construct a stress test-based evaluation platform called EQUATE (Evaluating Quantity Understanding Aptitude in Textual Entailment) (§5.4.2), which consists of five evaluation sets, each featuring different facets of quantitative reasoning in textual entailment (Table 5.12), including verbal reasoning with quantities, basic arithmetic computation, dealing with approximations and range comparisons. Given this evaluation platform, we study the ability of existing state-of-the-art NLI models to perform quantitative reasoning by benchmarking 9 published models on EQUATE. Our results show that most SOTA models are incapable of handling quantitative reasoning phenomena, instead relying on lexical cues for prediction. Additionally, we build a shallow semantic reasoning baseline for quantitative reasoning in NLI called Q-REAS, and evaluate its performance on EQUATE. Q-REAS is effective on synthetic test sets which require more quantity-based inference, but shows limited success on natural test sets which require deeper interaction between quantity-based and linguistic reasoning. The EQUATE evaluation framework highlights micro long tail quantitative reasoning phenomena that are still challenging, helping us identify areas to tackle to develop this skill better in NLU models.

5.4.2 Phenomena Selection from Task Knowledge

Unlike the previous case study, we do not rely on error analyses to select phenomena to focus our stress test construction on. Instead we turn to human knowledge of abilities required to perform quantitative reasoning. Our definition of "quantitative reasoning" draws from cognitive testing and education (Stafford, 1972; Ekstrom et al., 1976), which considers it a "verbal problem-solving ability". While inextricably linked to mathematics, it is an inclusive skill involving everyday language rather than a specialized lexicon. To excel at quantitative reasoning, one must possess a wide range of abilities such as interpreting quantities expressed in language, performing basic calculations, judging their accuracy, and justifying quantitative claims using verbal and numeric reasoning. These requirements show an interesting reciprocity: NLI naturally lends itself as a test bed for quantitative reasoning, which conversely, is important for NLI (Sammons et al., 2010; Clark, 2018).

Based on our knowledge of the spectrum of abilities required to perform quantitative reasoning, we select two key micro long tail phenomena (quantifiers and arithmetic reasoning), and construct

Phenomenon	Example
Arithmetic	**P:** Sharper faces charges in Arizona and California **H:** Sharper has been charged in two states
Ranges	**P:** Between 20 and 30 people were trapped in the casino **H:** Upto 30 people thought trapped in casino
Quantifiers	**P:** Poll: Obama over 50% in Florida **H:** New poll shows Obama ahead in Florida
Ordinals	**P:** Second-placed Nancy celebrated their 40th anniversary with a win **H:** Nancy stay second with a win
Approximation	**P:** Rwanda has dispatched 1917 soldiers **H:** Rwanda has dispatched some 1900 soldiers
Ratios	**P:** Londoners had the highest incidence of E. Coli bacteria (25%) **H:** 1 in 4 Londoners have E. Coli bacteria
Comparison	**P:** Treacherous currents took four lives on the Alabama Gulf coast **H:** Rip currents kill four in Alabama
Conversion	**P:** If the abuser has access to a gun, it increases chances of death by 500% **H:** Victim five times more likely to die if abuser is armed
Numeration	**P:** Eight suspects were arrested **H:** 8 suspects have been arrested
Implicit Quantities	**P:** The boat capsized two more times **H:** His sailboat capsized three times

Table 5.10: Examples of quantitative phenomena present in EQUATE.

synthetic stress tests for these. Additionally, we create three natural stress tests from real-world data, which focus solely on quantitative reasoning. Importantly, for natural test set creation, we sample *both* premise and hypothesis sentences from the wild, instead of having hypothesis sentences authored by crowdworkers. This reduces bias towards shallow expressions of complex micro long tail quantitative phenomena, while allowing us to construct test sets exhibiting how these phenomena are actually realized and used in everyday language.

In sum, EQUATE consists of five NLI test sets for quantitative reasoning. Three of these tests are natural tests, featuring language from real-world sources such as news articles and social media (RTE-Quant, NewsNLI, RedditNLI). We sample sentences containing quantities with numerical values, and consider an entailment pair to feature quantitative reasoning if it is at least one component of the overall reasoning required to determine the entailment label (but not necessarily the only reasoning component). Quantitative reasoning includes quantity

Source	Test Set	Size	Classes	Data Source	Annotation Source	Quantitative Phenomena
Natural	RTE-Quant	166	2	RTE2-RTE4	Experts	Arithmetic, Ranges, Quantifiers
	NewsNLI	968	2	CNN	Crowdworkers	Ordinals, Quantifiers, Arithmetic, Approximation, Magnitude, Ratios
	RedditNLI	250	3	Reddit	Experts	Range, Arithmetic, Approximation, Verbal
Synthetic	ST-Quant	7500	3	AQuA-RAT	Automatic	Quantifiers
	AwpNLI	722	2	Arithmetic Word Problems	Automatic	Arithmetic

Table 5.11: An overview of test sets included in EQUATE. RedditNLI and ST-Quant are framed as 3-class (entailment, neutral, contradiction) while RTE-Quant, NewsNLI and AwpNLI are 2-class (entails=yes/no). RTE 2-4 formulate entailment as a 2-way decision. We find that few news article headlines are contradictory, thus NewsNLI is similarly framed as a 2-way decision. For algebra word problems, substituting the wrong answer in the hypothesis necessarily creates a contradiction under the event coreference assumption de Marneffe et al. (2008), thus it is framed as a 2-way decision as well.

matching, quantity comparison, quantity conversion, arithmetic, qualitative processes, ordinality and quantifiers, quantity noun and adverb resolution (such as the quantities represented in *dozen*, *twice*, *teenagers*), as well as verbal reasoning with the quantity's textual context. For example, consider the sentence pair ⟨Obama cuts tax rate to 28%, Obama wants to cut tax rate to 28% as part of overhaul⟩. In addition to comparing the quantity (28% tax rate), we need to compare the contexts in which it has been used (cutting the rate vs wanting to cut the rate), to come up with the correct label. Table 5.10 presents some examples which demonstrate interesting quantitative phenomena that must be understood to label the pair correctly. It is important to note that we filter out sentence pairs which require only temporal reasoning, since specialized knowledge beyond knowledge of numbers, is needed to reason about time. These three test sets contain pairs which conflate some lexical and quantitative reasoning phenomena. In order to study some long tail quantitative reasoning phenomena in isolation, EQUATE further features two controlled synthetic tests (AwpNLI, ST-Quant), evaluating model ability to reason with quantifiers and perform simple arithmetic. Table 5.11 gives a brief overview of all five test sets, and provides some additional statistics and metadata about each set, while Appendix C presents some additional examples of annotated instances from EQUATE. As with our previous case study, we intend to jointly evaluate model performance on standard NLI datasets and the EQUATE benchmark, to evaluate model competence at quantitative reasoning in natural language understanding.

5.4.3 Constructing Stress Tests for EQUATE

RTE-Quant

This test set is constructed from the RTE sub-corpus for quantity entailment (Roy, 2017), originally drawn from the RTE2-RTE4 datasets (Dagan et al., 2006). The original sub-corpus conflates temporal and quantitative reasoning. We discarded pairs requiring temporal reasoning, obtaining a set of 166 entailment pairs.

NewsNLI

This test set is created from the CNN corpus of news articles with abstractive summaries (Hermann et al., 2015). We identify summary points with quantities, filtering out temporal expressions. For each summary point, the two most similar sentences from the article (according to Jaccard similarity) are chosen, flipping pairs where the premise begins with a first-person pronoun (e.g., ⟨"He had nine pears", "Bob had nine pears"⟩ becomes ⟨"Bob had nine pears", "He had nine pears"⟩). The top 50% of similar pairs are retained to avoid lexical overlap bias. We crowdsource annotations for a subset of this data from Amazon Mechanical Turk. Crowdworkers are shown two sentences and asked to determine whether the second sentence is definitely true, definitely false, or not inferable given the first. All crowdworkers are required to have an approval rate of 95% on at least 100 tasks and pass a qualification test. We collect 5 annotations per pair, and consider pairs with lowest token overlap between premise and hypothesis and least difference in premise-hypothesis lengths when stratified by entailment label. Top 1000 samples meeting these criteria form our final set. To validate crowdsourced labels, experts are asked to annotate 100 pairs. Crowdsourced gold labels match expert gold labels in 85% cases, while individual crowdworker labels match expert gold labels in 75.8%. Disagreements are manually resolved by experts and examples not featuring quantitiative reasoning are filtered, leaving a set of 968 samples.

RedditNLI

This test set is sourced from the popular social forum \reddit[3]. Since reasoning about quantities is important in domains like finance or economics, we scrape all headlines from the posts on \r\economics, considering titles that contain quantities and do not have meta-forum information. Titles appearing within three days of each other are clustered by Jaccard similarity, and the top 300 pairs are extracted. After filtering out nonsensical titles, such as concatenated stock prices, we are left with 250 sentence pairs. Similar to RTE, two expert annotators label these pairs, achieving a Cohen's kappa of 0.82. Disagreements are discussed to resolve final labels.

[3]According to the Reddit User Agreement, users grant Reddit the right to make their content available to other organizations or individuals.

ST-Quant

We include the numerical reasoning stress test from the previous case study (Naik et al., 2018) as one of our synthetic controlled test sets. This stress test consists of 7500 entailment pairs constructed from sentences in algebra word problems datasets (Ling et al., 2017). Focusing on quantifiers, it requires models to compare entities from hypothesis to the premise while incorporating quantifiers, but does not require them to perform the computation from the original algebra word problem (eg: ⟨"NHAI employs 100 men to build a highway of 2 km in 50 days working 8 hours a day","NHAI employs less than 700 men to build a highway of 2 km in 50 days working 8 hours a day"⟩).

AwpNLI

To evaluate arithmetic ability of NLI models, we repurpose data from arithmetic word problems (Roy and Roth, 2015). They have the following characteristic structure. First, they establish a world and optionally update its state. Then, a question is posed about the world. This structure forms the basis of our pair creation procedure. World building and update statements form the premise. A hypothesis template is generated by identifying modal/auxiliary verbs in the question, and subsequent verbs, which we call secondary verbs. We identify the agent and conjugate the secondary verb in present tense followed by the identified unit to form the final template (for example, the algebra word problem 'Gary had 73.0 dollars. He spent 55.0 dollars on a pet snake. How many dollars did Gary have left?' would generate the hypothesis template 'Agent(Gary) Verb(Has) Answer(18.0) Unit(dollars) left'). For every template, the correct guess is used to create an entailed hypothesis. Contradictory hypotheses are created by randomly sampling a wrong guess ($x \in \mathbb{Z}^+$ if correct guess is an integer, and $x \in \mathbb{R}^+$ if it is a real number) from a uniform distribution over an interval of 10 around the correct guess (or 5 for numbers less than 5), to identify plausible wrong guesses. We check for grammaticality, finding only 2% ungrammatical hypotheses, which are manually corrected leaving a set of 722 pairs.

5.4.4 Experiments and Analysis

Neural NLI Models

We evaluate the performance of the following 9 neural NLI models, and 2 additional non-neural baseline models:

1. **Majority Class (MAJ):** Simple baseline that always predicts the majority class in test set.
2. **Hypothesis-Only (HYP):** FastText classifier (Joulin et al., 2017) trained to predict entailment labels from hypothesis sentences only (Gururangan et al., 2018).
3. **ALIGN:** A bag-of-words alignment model inspired by MacCartney (2009), which bases the entailment prediction on lexical overlap between premise and hypothesis, utilizing

RTE-QUANT
P: After the deal closes, Teva will generate sales of about $ 7 billion a year, the company said. **H**: Teva earns $ 7 billion a year.
AWP-NLI
P: Each of farmer Cunningham's 6048 lambs is either black or white and there are 193 white ones. **H**: 5855 of Farmer Cunningham's lambs are black.
NEWSNLI
P: Emmanuel Miller, 16, and Zachary Watson, 17, are charged as adults, police said. **H**: Two teen suspects charged as adults.
REDDITNLI
P: Oxfam says richest one percent to own more than rest by 2016. **H**: Richest 1% To Own More Than Half Worlds Wealth By 2016 Oxfam.

Table 5.12: Examples from evaluation sets in EQUATE.

Levenshtein edit distance and weighting term importance by part-of-speech tags. The accuracy of our re-implementation of this model on RTE-3 test is 61.12%, comparable to the reported average model performance of 62.4% on the RTE challenge.

4. **CBOW:** A simple bag-of-embeddings sentence representation model (Williams et al., 2018), using GloVe word embeddings (Pennington et al., 2014).

5. **BiLSTM:** The simple BiLSTM model described by Williams et al. (2018), which forms sentence representations by averaging the states of a BiLSTM over the words in the sentence.

6. **Chen (CH):** A stacked BiLSTM-RNN model with shortcut connections and character-CNN embeddings Chen et al. (2017b).

7. **InferSent:** A single-layer BiLSTM-RNN model with max-pooling (Conneau et al., 2017), which is trained to learn robust universal sentence representations that transfer well across several inference tasks.

8. **SSEN:** A stacked BiLSTM-RNN model with shortcut connections (Nie and Bansal, 2017), which was the best-performing model on the RepEval shared task on MultiNLI (Nangia et al., 2017b).

9. **ESIM:** Sequential inference model proposed by Chen et al. (2017a) which uses BiLSTMs with an attention mechanism.

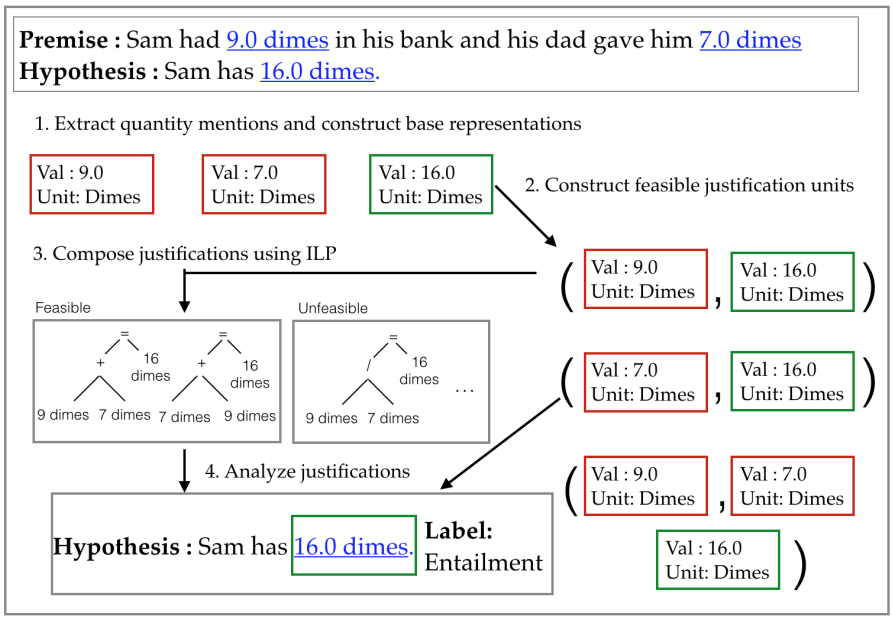


Figure 5.3: Overview of the Q-REAS baseline.

10. **OpenAI GPT:** Transformer-based language model (Vaswani et al., 2017), with finetuning on NLI (Radford et al., 2018).

11. **BERT:** Transformer-based language model (Vaswani et al., 2017), with cloze-style and next-sentence prediction objectives, and finetuning on NLI Devlin et al. (2019).

In addition to these models, we also evaluate the performance of Q-REAS, our shallow symbolic reasoning baseline described in the next section.

Q-REAS: A Shallow Symbolic Reasoning Baseline

Figure 5.3 gives an overview of the structure of our Q-REAS baseline for quantitative reasoning. This model manipulates quantity representations symbolically to make entailment decisions, and is intended to serve as a strong heuristic baseline for numerical reasoning on the EQUATE benchmark. This model has four main stages:

1. Quantity mentions are extracted and parsed into semantic representations called NUMSETS (Quantity Segmenter, Quantity Parser).

2. Compatible pairs of NUMSETS from premise and hypothesis sentences are formed (Quantity Pruner).

3. Compatible NUMSET pairs are composed to form *justifications* (Quantity Composition).

4. Justifications are analyzed to make the final entailment decisions (Global Reasoner).

INPUT	
P_c	Set of "compatible" single-valued premise quantities
P_r	Set of "compatible" range-valued premise quantities
H	Hypothesis quantity
O	Operator set $\{+, -, *, /, =, \cap, \cup, \setminus, \subseteq\}$
L	Length of equation to be generated
SL	Symbol list ($P_c \cup P_r \cup H \cup O$)
TL	Type list (set of types from P_c, P_r, H)
N	Length of symbol list
K	Index of first range quantity in symbol list
M	Index of first operator in symbol list
OUTPUT	
e_i	Index of symbol assigned to i^{th} position in postfix equation
VARIABLES	
x_i	Main ILP variable for position i
c_i	Indicator variable: is e_i a single value?
r_i	Indicator variable: is e_i a range?
o_i	Indicator variable: is e_i an operator?
d_i	Stack depth of e_i
t_i	Type index for e_i

Table 5.13: Input, output and variable definitions for the Integer Linear Programming (ILP) framework used for quantity composition.

Quantity Segmenter

We follow Barwise and Cooper (1981) in defining quantities as having a number, unit, and an optional approximator. Quantity mentions are identified as least ancestor noun phrases from the constituency parse of the sentence containing cardinal numbers.

Quantity Parser

The quantity parser constructs a grounded representation for each quantity mention in the premise or hypothesis, henceforth known as a NUMSET (note that a NUMSET may be a composition of other NUMSETS). A NUMSET is a tuple (val, unit, ent, adj, loc, verb, freq, flux)[4] where:

1. val $\in [\mathbb{R}, \mathbb{R}]$: quantity value represented as a range
2. unit $\in S$: unit noun associated with the quantity
3. ent $\in S^{\phi}$: entity noun associated with the unit (e.g., '*donations* worth 100$')

[4] As in Koncel-Kedziorski et al. (2015), S denotes all possible spans in the sentence, ϕ represents the empty span, and S^{ϕ}=$S \cup \phi$

4. adj $\in S^{\phi}$: adjective associated with unit if any, extracted as governing verb linked to entity by an *amod* relation

5. loc $\subseteq S^{\phi}$: location of the unit (e.g.,'in the bag'), extracted as the prepositional phrase attached to the quantity and containing noun phrase

6. verb $\in S^{\phi}$: action verb associated with the quantity, extracted as governing verb linked to entity by *dobj* or *nsubj* relation

7. freq $\subseteq S^{\phi}$: if quantity recurs, extracted using keywords *per* and *every* (e.g, 'per hour')

8. flux $\in$ {increase to, increase from, decrease to, decrease from}$^{\phi}$: if quantity is in a state of flux, extracted using a gazetteer: *increasing*, *rising*, *rose*, *decreasing*, *falling*, *fell*, *drop*

To extract **values** for a quantity, we extract cardinal numbers, recording contiguity. We normalize the number by remove ","s, converting written numbers to floats and deciding the numerical values (for example hundred fifty eight thousand is 158000, two fifty eight is 258, 3.74m is 3740000 etc.). If cardinal numbers are non-adjacent, we look for an explicitly mentioned range such as 'to' and 'between'. We also handle simple ratios such as quarter, half etc, and extract bounds (e.g., *fewer than 10 apples* is parsed to $[-\infty, 10]$ apples.)

To extract **units**, we examine tokens adjacent to cardinal numbers in the quantity mention and identify known units. If no known units are found, we assign the token in a *numerical modifier* relationship with the cardinal number, else we assign the nearest noun to the cardinal number as the unit. A quantity is determined to be **approximate** if the word in an *adverbial modifier* relation with the cardinal number appears in a gazetteer[5]. If approximate, range is extended to (+/-)2% of the current value.

Quantity Pruner

The pruner constructs "compatible" premise-hypothesis NUMSET pairs. Consider the pair "Insurgents killed *7 U.S. soldiers*, set off a car bomb that killed *four Iraqi policemen*" and "*7 US soldiers* were killed, and *at least 10 Iraqis* died". Our parser extracts NUMSETS corresponding to *"four Iraqi policemen"* and *"7 US soldiers"* from premise and hypothesis respectively. But these NUMSETSshould not be compared as they involve different units. The pruner discards such incompatible pairs. Heuristics to identify unit-compatible NUMSETpairs include three cases: (i) direct string match, (ii) synonymy/hypernymy relations from WordNet, and (iii) one unit is a nationality/job and the other unit is synonymous with person, a heuristic also used by Roy (2017). Lists of jobs and nationalities are scraped from Wikipedia.

[5]roughly, approximately, about, nearly, roundabout, around, circa, almost, approaching, pushing, more or less, in the neighborhood of, in the region of, on the order of,something like, give or take (a few), near to, close to, in the ballpark of

Definitional Constraints	
Range restriction	$x_i < K$ or $x_i = M - 1$ for $i \in [0, L-1]$ if $c_i = 1$
	$x_i \geq K$ and $x_i < M$ for $i \in [0, L-1]$ if $r_i = 1$
	$x_i \geq M$ for $i \in [0, L-1]$ if $o_i = 1$
Uniqueness	$c_i + r_i + o_i = 1$ for $i \in [0, L-1]$
Stack definition	$d_0 = 0$ (Stack depth initialization)
	$d_i = d_{i-1} - 2o_i + 1$ for $i \in [0, L-1]$ (Stack depth update)
Syntactic Constraints	
First two operands	$c_0 + r_0 = 1$ and $c_1 + r_1 = 1$
Last operator	$x_{L-1} \geq N - 1$ (Last operator should be one of $\{=, \subseteq\}$)
Last operand	$x_{L-2} = M - 1$ (Last operand should be hypothesis quantity)
Other operators	$x_i \leq N - 2$ for $i \in [0, L-3]$ if $o_i = 1$
Other operands	$x_i < K$ for $i \in [0, L-3]$ if $c_i = 1$
	$x_i < M$ for $i \in [0, L-3]$ if $r_i = 1$
Empty stack	$d_{L-1} = 0$ (Non-empty stack indicates invalid postfix expression)
Premise usage	$x_i \neq x_j$ for $i, j \in [0, L-1]$ if $o_i \neq 1, o_j \neq 1$
Operand Access	
Right operand	$op2(x_i) = x_{i-1}$ for $i \in [0, L-1]$ such that $o_i = 1$
Left operand	$op1(x_i) = x_l$ for $i, l \in [0, L-1]$ where $o_i = 1$ and l is the largest index such that $l \leq (i-2)$ and $d_l = d_i$

Table 5.14: Mathematical validity constraint definitions for the ILP framework. Functions $op1()$ and $op2()$ return the left and right operands for an operator respectively. Variables defined in Table 5.13.

Quantity Composition

The composition module detects whether a hypothesis NUMSET is justified by composing "compatible" premise NUMSETS. For example, consider the pair "I had *3 apples* but gave *one* to my brother" and "I have *two apples*". Here, the premise NUMSETS P_1 ("*3 apples*") and P_2 ("*one apple*") must be composed to deduce that the hypothesis NUMSET H_1 ("*2 apples*") is justified. Our framework accomplishes this by generating postfix arithmetic equations from premise NUMSETS, that justify the hypothesis NUMSET. Note that arithmetic equations differ from algebraic equations in that they do *not* contain unknown variables. Direct comparisons between NUMSETS are incorporated by adding "=" as an operator. In this example, the expression $< P_1, P_2, -, H_1, =>$ will be generated.

The set of possible equations is exponential in number of NUMSETS, making exhaustive generation intractable. But a large number of equations are invalid as they violate constraints such as unit consistency. Thus, our framework uses integer linear programming (ILP) to constrain the equation space. It is inspired by prior work on algebra word problems (Koncel-Kedziorski et al., 2015), with the following key differences:

1. **Arithmetic equations:** We focus on arithmetic equations instead of algebraic ones.

2. **Range arithmetic:** Quantitative reasoning involves ranges, which are handled by represent-

Type Consistency Constraints	
Type assignment	$t_i = TL[k]$ for $i \in [0, L-1]$ if $c_i + r_i = 1$ and $type(SL_i) = k$
Two type match	$t_i = t_a = t_b$ for $i \in [0, L-1]$ such that $o_i = 1, x_i \in \{+, -, *, /, =, \cap, \cup, \setminus, \subseteq\}, a = op1(x_i), b = op2(x_i)$
One type match	$t_i \in \{t_a, t_b\}, t_a \neq t_b$ for $i \in [0, L-1]$ such that $o_i = 1, x_i = *, a = op1(x_i), b = op2(x_i)$ $t_i = t_a \neq t_b$ for $i \in [0, L-1]$ such that $o_i = 1, x_i = /, a = op1(x_i), b = op2(x_i)$
Operator Consistency Constraints	
Arithmetic operators	$c_a = c_b = 1$ for $i \in [0, L-1]$ such that $o_i = 1, x_i \in \{+, -, *, /, =\}, a = op1(x_i), b = op2(x_i)$
Range operators	$r_a = r_b = 1$ for $i \in [0, L-1]$ such that $o_i = 1, x_i \in \{\cap, \cup, \setminus\}, a = op1(x_i), b = op2(x_i)$ $r_b = 1$ for $i \in [0, L-1]$ such that $o_i = 1, x_i = \subseteq, b = op2(x_i)$

Table 5.15: Linguistic consistency constraint definitions for the ILP framework. Functions $op1()$ and $op2()$ return the left and right operands for an operator respectively. Variables defined in Table 5.13.

ing them as endpoint-inclusive intervals and adding the four operators ($\cup, \cap, \setminus, \subseteq$).

3. **Hypothesis quantity-driven:** We optimize an ILP model for each hypothesis NUMSET because a sentence pair is marked "entailment" iff every hypothesis quantity is justified.

Table 5.13 describes the variables used in our ILP problem formulation. We impose the following types of ILP constraints:

1. **Definitional Constraints:** Ensure that ILP variables take on valid values by constraining initialization, range, and update.

2. **Syntactic Constraints:** Assure syntactic validity of generated postfix expressions by limiting operator-operand ordering.

3. **Operand Access:** Simulate stack-based evaluation correctly by choosing correct operator-operand assignments.

4. **Type Consistency:** Ensure that all operations are type-compatible.

5. **Operator Consistency:** Force range operators to have range operands and mathematical operators to have single-valued operands.

Algorithm 1 PredictEntailmentLabel(P, H, C, E)

Input: Premise quantities P, Hypothesis quantities H, Compatible pairs C, Equations E
Output: Entailment label $l \in$ { e, c, n }

if $C = \emptyset$ **then return** n
end if
$J \leftarrow \emptyset$
$L \leftarrow []$
for $q_h \in H$ **do**
    $J_h \leftarrow \{q_p \mid q_p \in P, (q_p, q_h) \in C\}$
    $J \leftarrow J \cup \{(q_h, J_h)\}$
    $L \leftarrow L + [false]$
end for
for $(q_h, J_h) \in J$ **do**
    if $J_h = \emptyset$ **then return** n
    end if
    for $q_p \in J_h$ **do**
        $s \leftarrow$ MaxSimilarityClass(q_p, q_h)
        if $s = e$ **then**
            if ValueMatch(q_p, q_h) **then**
                $L[q_h] = true$
            end if
            if !ValueMatch(q_p, q_h) **then**
                $L[q_h] = false$
            end if
        end if
        if $s = c$ **then**
            if ValueMatch(q_p, q_h) **then**
                $L[q_h] = c$
            end if
        end if
    end for
end for
for $q_h \in H$ **do**
    $E_q \leftarrow \{e_i \in E \mid hyp(e_i) = q_h\}$
    if $E_q \neq \emptyset$ **then**
        $L[q_h] = true$
    end if
end for
if $c \in L$ **then return** c
end if
if count$(L, true)$ = len(L) **then return** e
return n
end if

M \ D	RTE-Q	Δ	NewsNLI	Δ	RedditNLI	Δ	ST-Q	Δ	AWPNLI	Δ	Nat. Avg.	Δ	Synth. Avg.	Δ	All Avg.	Δ
MAJ	57.8	0.0	50.7	0.0	**58.4**	**0.0**	33.3	0.0	50.0	0.0	+0.0		+0.0		+0.0	
HYP	49.4	-8.4	52.5	+1.8	40.8	-17.6	31.2	-2.1	50.1	+0.1	-8.1		-1.0		-5.2	
ALIGN	62.1	+4.3	56.0	+5.3	34.8	-23.6	22.6	-10.7	47.2	-2.8	-4.7		-6.8		-5.5	
CBOW	47.0	-10.8	61.8	+11.1	42.4	-16.0	30.2	-3.1	50.7	+0.7	-5.2		-1.2		-3.6	
BiLSTM	51.2	-6.6	63.3	+12.6	50.8	-7.6	31.2	-2.1	50.7	+0.7	-0.5		-0.7		-0.6	
CH	54.2	-3.6	64.0	+13.3	55.2	-3.2	30.3	-3.0	50.7	+0.7	+2.2		-1.2		+0.9	
InferSent	66.3	+8.5	65.3	+14.6	29.6	-28.8	28.8	-4.5	50.7	+0.7	-1.9		-1.9		-1.9	
SSEN	58.4	+0.6	65.1	+14.4	49.2	-9.2	28.4	-4.9	50.7	+0.7	+1.9		-2.1		+0.3	
ESIM	54.8	-3.0	62.0	+11.3	45.6	-12.8	21.8	-11.5	50.1	+0.1	-1.5		-5.7		-3.2	
GPT	**68.1**	**+10.3**	72.2	+21.5	52.4	-6.0	36.4	+3.1	50.0	+0.0	+8.6		+1.6		+5.8	
BERT	57.2	-0.6	**72.8**	**+22.1**	49.6	-8.8	36.9	+3.6	42.2	-7.8	+4.2		-2.1		+1.7	
Q-REAS	56.6	-1.2	61.1	+10.4	50.8	-7.6	**63.3**	**+30**	**71.5**	**+21.5**	+0.5		+25.8		**+10.6**	

Table 5.16: Accuracies(%) of 9 NLI Models on five tests for quantitiative reasoning in entailment. M and D represent *models* and *datasets* respectively. Δ captures improvement over majority-class baseline for a dataset. Column Nat.Avg. reports the average accuracy(%) of each model across 3 evaluation sets constructed from natural sources (RTE-Quant, NewsNLI, RedditNLI), whereas Synth.Avg. reports the average accuracy(%) on 2 synthetic evaluation sets (ST-Quant, AwpNLI). Column Avg. represents the average accuracy(%) of each model across all 5 evaluation sets in EQUATE.

Definitional, syntactic, and operand access constraints ensure mathematical validity while type and operator consistency constraints add linguistic consistency. Constraint formulations are provided in Tables 5.14 and 5.15. We limit tree depth to 3 and retrieve a maximum of 50 solutions per hypothesis NUMSET, then solve to determine whether the equation is mathematically correct. We discard equations that use invalid operations (division by 0) or add unnecessary complexity (multiplication/ division by 1). The remaining equations are considered plausible justifications.

Global Reasoner

The global reasoner predicts the final entailment label as shown in Algorithm 1. MaxSimilarity-Class() takes two quantities and returns a probability distribution over entailment labels based on unit match. Similarly, ValueMatch() detects whether two quantities match in value (this function can also handle ranges), on the assumption that every NUMSET in the hypothesis *has* to be justified to predict the label to be entailment. Note that this is a necessary but not sufficient condition for entailment. Consider the example, ⟨'Sam believed Joan had 5 apples', 'Joan had 5 apples'⟩. The hypothesis quantities of 5 apples is justified but is not a sufficient condition for entailment.

Model Performance on EQUATE

Table 5.16 presents the results of all models on EQUATE. Table 5.17 presents classification accuracies of all neural models used on the matched development set of MultiNLI. All models, except Q-REAS (which requires no supervision) are trained on MultiNLI (Williams et al., 2018). The only data sources used by Q-REAS are WordNet and lists from Wikipedia. From these tables,

Model	MultiNLI Dev
Hyp Only	53.18%
ALIGN	45.0%
CBOW	63.5%
BiLSTM	70.2%
Chen	73.7%
NB	74.2%
InferSent	70.3%
ESIM	76.2%
OpenAI GPT	81.35%
BERT	83.8%

Table 5.17: Performance of all baseline models used in the paper on the matched development set of MultiNLI. These scores are very close to the numbers reported by the original publications, affirming the correctness of our baseline setup.

we observe that neural models, particularly OpenAI GPT excel at verbal aspects of quantitative reasoning (RTE-Quant, NewsNLI), whereas our symbolic baseline Q-REAS excels at numerical aspects (ST-Quant, AwpNLI).

Analyzing Neural Model Performance on NewsNLI

To tease apart contributory effects of numerical and verbal reasoning in natural data, we explore model performance on NewsNLI. We extract all entailed pairs where a quantity appears in both premise and hypothesis, and perturb the quantity in the hypothesis generating contradictory pairs. For example, the pair ⟨'In addition to 79 fatalities, some 170 passengers were injured.'⟩ 'The crash took the lives of 79 people and injured some 170', 'entailment' is changed to ⟨'In addition to 79 fatalities, some 170 passengers were injured.', 'The crash took the lives of 80 people and injured some 170', 'contradiction'⟩, assuming scalar implicature and event coreference. Our perturbed test set contains 218 pairs. On this set, GPT, the best-performing neural model on EQUATE, achieves an accuracy of 51.18%, as compared to 72.04% on the unperturbed set, suggesting the model relies on verbal cues rather than numerical reasoning. In comparison, Q-REAS achieves an accuracy of 98.1% on the perturbed set, compared to 75.36% on the unperturbed set, highlighting reliance on quantities rather than verbal information. Closer examination reveals that GPT switches to predicting the 'neutral' category for perturbed samples instead of entailment, accounting for 42.7% of its errors, possibly symptomatic of lexical bias issues (Naik et al., 2018; McCoy et al., 2019).

Identifying Hard Quantitative Phenomena

A key advantage of stress test-based evaluation frameworks like EQUATE is the possibility of identifying and isolating micro long tail phenomena that models are unable to solve, which can offer clear directions for future work. For quantitative reasoning, we identify such hard phenomena which cannot be addressed by simple quantity comparison, by sampling 100 errors made by Q-REAS on each test in EQUATE. Our analysis of causes for error suggest the following avenues for future research on quantitative reasoning:

1. **Multi-step numerical-verbal reasoning:** Models do not perform well on examples requiring interleaved verbal and quantitative reasoning, especially multi-step deduction. Consider the pair ⟨"Two people were injured in the attack", "Two people perpetrated the attack"⟩. Quantities "two people" and "two people" are unit-compatible, but must not be compared. Another example is the NewsNLI entailment pair in Table 5.12. This pair requires us to identify that 16 and 17 refer to Emmanuel and Zachary's ages (quantitative), deduce that this implies they are teenagers (verbal) and finally count them (quantitative) to get the hypothesis quantity "two teens". Numbers and language are intricately interleaved and developing a reasoner capable of handling such complex interplay is challenging.

2. **Lexical inference:** Lack of real world knowledge causes errors in identifying quantities and valid comparisons. Errors include mapping abbreviations to correct units ("m" to "meters"), detecting part-whole coreference ("seats" can be used to refer to "buses"), and resolving hypernymy/hyponymy ("young men" to "boys").

3. **Inferring underspecified quantities:** Quantity attributes can be implicitly specified, requiring inference to generate a complete representation. Consider "A mortar attack killed four people and injured 80". A system must infer that the quantity "80" refers to people. On RTE-Quant, 20% of such cases stem from zero anaphora, a hard problem in coreference resolution.

4. **Arithmetic comparison limitations:** These examples require composition between incompatible quantities. For example, consider ⟨"There were 3 birds and 6 nests", "There were 3 more nests than birds"⟩. To correctly label this pair "3 birds" and "6 nests" must be composed.

Final observations from case study 2:

Our analyses of performance of state-of-the-art neural NLI models on MultiNLI and EQUATE, our proposed evaluation benchmark for quantitative reasoning in NLI leave us with the following observations:

- Evaluating model ability to tackle a specific skill using non-identically distributed test-only datasets, is a more effective way of estimating true model performance on that skill

(quantitative reasoning in this study). This evaluation strategy is immune to high-frequency spurious label correlations that may be present in training/test sets in IID or PAID evaluation paradigms.

- Skill-focused testing allows us to isolate key micro long tail phenomena that are required for reasoning, but are not handled well by current models. Isolating these phenomena is extremely useful in determining new avenues for further research.

5.5 Discussion and Related Work

There have been many critiques of contemporary evaluation paradigms, especially the identically distributed evaluation paradigm. For example, Smith (2012) discuss dangers of community-wide "overfitting" to benchmark datasets and emphasize the need to correlate model errors to well-defined linguistic phenomena to understand specific model strengths and weaknesses. Over the years, several alternatives to the identically distributed evaluation paradigm have been proposed, which can broadly be categorized into the following major categories.

5.5.1 Adversarial Evaluation

Adversarial evaluation schemes primarily focus on evaluating robustness of models on various NLP tasks using adversarial examples. Adversarial examples are created by applying minimal *label-preserving* perturbations to existing datasets. These examples *attack* NLP models, with perturbations acting as distractions, and reveal spurious correlations and biases that models rely on, but are not relevant for the task. Unlike computer vision in which such perturbed examples can be created by introducing minimal amounts of noise (Szegedy et al., 2014; Goodfellow et al., 2015), adversarial perturbation for discrete sequences is a harder problem. Despite this difficulty, recent years have seen the emergence of more work on adversarial example construction for text data (Goodfellow et al., 2015; Papernot et al., 2016; Samanta and Mehta, 2017; Sakaguchi et al., 2017; Liang et al., 2018; Ebrahimi et al., 2018; Gao et al., 2018; Iyyer et al., 2018; Ribeiro et al., 2018; Wallace et al., 2019a).

Jia and Liang (2017) were among the earliest to explore the use of adversarial examples for evaluation in natural language understanding, focusing on the task of machine reading comprehension. For this task, they showed that concatenating a distractor sentence (also called a concatenative adversary) to the context paragraph was enough to distract state-of-the-art models which were no longer able to extract the correct answer span from the passage. This sparked extensive work on developing adversarial evaluation schemes for specific NLP tasks (e.g., Belinkov and Bisk (2018) for machine translation), as well as on developing universal adversarial evaluation schemes that apply across a range of NLP tasks (Wallace et al., 2019a). In particular, the BIBINLP (Build It Break It, The Language Edition) shared task Ettinger et al. (2017) played a huge role in spurring the development of automated/manually created adversarial examples for a host of tasks and models.

Our proposed stress test evaluation paradigm adds a focus on specific sets of linguistic phenomena in addition to testing robustness of NLU models.

In addition to developing adversarial evaluation sets, several recent works have proposed adversarial collection strategies to improve the dataset annotation process. These strategies can be broadly divided into two categories: (i) techniques that filter out examples which are easily answered by SOTA models (Dua et al., 2019; Dasigi et al., 2019), and (ii) techniques that use humans or SOTA models in the loop, to generate adversarial inputs (Zellers et al., 2018, 2019; Nie et al., 2019; Wallace et al., 2019b). Though such datasets still perform identically distributed evaluation, incorporating adversarial filtering or example construction partly reduces biases during the collection process. These approaches however closely tie the dataset construction process to the capabilities of the models used for filtering/construction, and may end up biasing the dataset towards the quirks of those models (Zellers et al., 2019). We avoid such procedures during stress test construction.

5.5.2 Challenge Sets

Unlike adversarial evaluation paradigms, challenge sets focus on evaluating the performance of NLP models on *specific linguistic phenomena*. Such challenge sets have also gained prominence in NLP, and their phenomenon focus makes them better-suited for long tail evaluation. Concurrent to our work on developing stress tests for natural language inference, Glockner et al. (2018) and McCoy et al. (2019) also develop challenge sets for NLI focused on various phenomena such as hypernymy, co-hyponymy, lexical overlap, etc. Challenge sets have also been developed for coreference resolution (Levesque, 2014), question answering (Clark et al., 2018), machine translation (Isabelle et al., 2017; Burlot and Yvon, 2017; Bawden et al., 2018), and a host of other tasks. Interestingly, many challenge sets follow a minimal edit construction strategy, analogous to adversarial example construction. Minimal-edit challenge sets have been constructed for machine translation (Sennrich, 2017), language modeling (Marvin and Linzen, 2018; Warstadt et al., 2020) and social bias detection (Rudinger et al., 2018; Zhao et al., 2018; Lu et al., 2018), among other tasks. Despite using similar strategies, challenge sets maintain a *phenomenon* focus.

Our proposed stress test evaluation paradigm fits into this category. Stress tests can be thought of as non-identically distributed test-only challenge sets. Most similar to our proposed paradigm is the recent work by Ribeiro et al. (2020) on behavioral testing of NLP models. Their work introduces CHECKLIST, a task-agnostic framework to test NLP models on a variety of *general linguistic capabilities* using various *test types*. Both works share the same underlying principles: borrowing the concept of unit testing/behavioral testing from software engineering and instantiating it as an NLP evaluation framework. However, their framework is task-agnostic and covers a broad range of basic linguistic capabilities: Vocabulary+POS (important words or word types for the task), Taxonomy (synonyms, antonyms, etc), Robustness (to typos, irrelevant changes, etc), NER (appropriately understanding named entities), Fairness, Temporal (understanding order of events),

Negation, Coreference, Semantic Role Labeling (understanding roles such as agent, object, etc), and Logic (ability to handle symmetry, consistency, and conjunctions). In addition, they provide three test types: MFT (Minimum Functionality Test), INV (Invariance Test), and DIR (Directional Expectation Test). Their work shows similar results: using behavioral testing, they are able to isolate "bugs", i.e., cases where NLP models break down due to their over-reliance on spurious correlations, which are not visible through identically distributed evaluation.

5.5.3 Counterfactual Evaluation/Contrast Sets

In addition to adversarial evaluation and challenge sets, an interesting recent direction is work on generating minimal-edit examples to close gaps in existing test sets, caused by sampling or annotator biases. Towards this end, Kaushik et al. (2019) and Gardner et al. (2020) propose to minimally perturb test instances in such a way that *the label changes*, creating what they call counterfactual evaluation sets and contrast sets respectively. Kaushik et al. (2019) obtain these minimally edited pairs using crowdsourcing, while Gardner et al. (2020) rely on experts to construct these examples. An important thing to note is that this work also does not focus on specific phenomena, making it conceptually similar to adversarial evaluation (with the exception that minimal edits are label-changing instead of label-preserving).

5.6 Conclusion

In this chapter, we proposed an evaluation paradigm to better examine the performance of models on micro long tail phenomena: stress tests. We defined stress tests as non-identically distributed test-only datasets focused on measuring model ability on a single linguistic phenomenon, or a small set of related phenomena. Our aim in creating these stress tests was to get a better picture of true model performance on micro long tail phenomena of interest, as well as identify strengths and weaknesses of models to obtain actionable insights about micro long tail phenomena that are not handled by models. Using stress tests, we performed two case studies on natural language inference and quantitative reasoning in NLI. Both case studies demonstrated that existing identically distributed evaluation gives over-optimistic estimates of model ability to truly understand language. Our stress tests were more effective in unveiling key weaknesses in current NLI models such as inability to handle lexical relations, and multiple meaning systems (language and numbers). We hope that stress-based evaluation is used to supplement existing evaluation paradigms, to obtain more accurate model performance, especially on micro long tail phenomena and isolate more phenomena of interest that contemporary models fail on, to identify more avenues for future research.

CHAPTER 6

Conclusion and Future Directions

6.1 Summary of Contributions

To improve model performance and evaluation on the long tail in language understanding, this thesis explored the applicability of both existing and newly proposed transfer learning methods, in addition to presenting a new evaluation paradigm. We first discussed a two-level (macro and micro) conceptualization of the long tail, and highlighted three research questions:

- How can we best adapt benchmark-trained models across macro long tail dimensions?
- How can we best equip benchmark-trained models to handle micro long tail phenomena?
- How can we comprehensively evaluate model performance on the long tail?

Through a series of case studies, we tried to address these research questions, while building up a set of best practices that could potentially apply to newer long tail settings. We briefly summarize the contributions of this thesis, and enumerate the set of best practices observed from our studies.

6.1.1 Dataset Contributions

This thesis contributes several new, interesting datasets focused on the long tail, which may be useful for future work in this area:[1]

- **MTSamples:** A test-only dataset consisting of clinical narratives from three different specialties, annotated with entity and event spans (Chapter 3).
- **TDDiscourse:** A dataset augmenting an existing temporal ordering benchmark with additional temporal relation annotations for long-distance event pairs (Chapter 4).

[1] Note that we do not list the dataset of clinical conversations, since the conversations are sourced from a proprietary dataset owned by Abridge, Inc., and we do not have permission to distribute it.

- **NLI Stress Tests:** A stress-based evaluation platform for the task of natural language inference (NLI), consisting of test sets focused on six phenomena of interest grouped into three broad categories (Chapter 5).
- **EQUATE:** An NLI-based evaluation platform for quantitative reasoning consisting of five test sets, both synthetic and natural, covering a broad range of quantitative phenomena (Chapter 5).

6.1.2 Modeling Contributions

This thesis also contributes several new transfer learning methods, which can be applied to long tail settings beyond the ones explored in this work:

- **Likelihood-based Instance Weighting (LIW):** An unsupervised adaptation method from the instance weighting hybrid category, which uses language model likelihoods to estimate source-target similarity and compute source instance weights (Chapter 3). Though our work primarily establishes its utility for clinical domains (both narratives and conversations), the method itself is general enough to be applicable to any long tail domain.
- **Domain-Aware Query Sampling (DAQ):** A domain-awareness criterion based on embedding similarity that improves data efficiency of active learning data-centric methods in an adaptation setting (Chapter 3). We validate its utility for both clinical and literary narratives, but this method is again general enough to be applicable to any long tail domain.
- **Neural-ILP Fusion for Temporal Ordering:** A loss augmentation method from the model-centric category, which uses a structured support vector machine (SSVM) formulation to incorporate predefined heuristics as integer linear programming (ILP) constraints during neural model training. The SSVM framework is general enough to be applicable to other neural architectures beyond the one explored in our case study (BiLSTMs).

6.1.3 Methodological Contributions and Recommendations

Finally, this thesis makes the following methodological contributions and recommendations:

- **Updated taxonomy of adaptation methods:** We provide updated version of the adaptation method taxonomy from Ramponi and Plank (2020), by extending it to cover pre-neural and supervised adaptation methods.
- **Stress testing paradigm:** We propose a new evaluation paradigm for the long tail called *stress testing*, which calls for supplementary evaluation of models on non-identically distributed phenomenon-focused test-only datasets. We also show the utility of this paradigm in identifying micro long tail phenomena that existing state-of-the-art models are unable to handle.
- **Best practice recommendations:** Through a series of case studies, we also aggregate a set of best practices (focused on information extraction tasks) that may be useful for better-informed selection of transfer methods, when applied to a new long tail setting:

1. In unsupervised settings, promising method categories for high-expertise narrative domains include loss augmentation, pseudo labeling and likelihood-based instance weighting. Loss augmentation methods work best when spans to be extracted contain highly technical vocabulary, otherwise pseudo-labeling methods are stronger. LIW works best for extraction of span types that are new within the target domain.
2. In unsupervised settings, pretraining appears to be the most promising method category for high-expertise non-narrative domains. Note that since the domain we experiment with is doctor-patient conversations (i.e., expert-novice setting), this observation could vary slightly in when dealing with non-narrative domains with higher technical content (e.g., Ubuntu IRC chats).
3. In a limited supervision setting, pretrained language models are extremely data-efficient and do not benefit much from active learning methods.
4. The term vocabulary overlap (TVO) metric, despite limited nuance, can strongly predict potential performance improvements/drops for most method categories.
5. When comparing performance of multiple transfer methods, going beyond overall scores can reveal particular strengths and weaknesses. We find the following analyses especially helpful: (i) Correlating performance changes with source-target distance, (ii) Analyzing performance on various kinds of linguistic shifts (e.g., lexical, semantic, etc.), and (iii) Qualitative analyses of method-specific success and error cases.

6.2 Limitations of this Thesis

Despite our best efforts to perform a broad-coverage set of case studies, there are some aspects that remain missing or under-explored, and must be addressed by future work:

Under-Explored Macro Dimensions: Throughout our work, we have studied a varied set of domains and adaptation settings. However, all our experiments have been limited to the same language (English), and the same set of tasks (text classification and sequence labeling). Given the massive space of possible choices under all four macro dimensions, we made these fixed choices to carve out a smaller subspace that we could feasibly explore within the scope of this thesis. This necessarily comes at a cost, it is unclear how well our observations and best practice recommendations would hold for non-English languages and other tasks. Case studies similar in nature to the ones presented in this thesis would need to be carried out with other languages and tasks to establish the utility of our recommendations, and we leave this as an open question to future work.

Missing Macro Dimensions: A second limitation of our work arises from our choice of specific dimensions to focus on at the macro-level. Again, we had to select a subset of possible dimensions to carve out a feasible experiment space, but this selection leaves out some potential dimensions, which are slightly out of scope given our focus on text understanding. One major macro dimension dropped from our work is modality, primarily due to our focus on text corpora. However, this

leads to two keys issues. Not considering other modalities may cause sidelining/missing out on languages that are never represented in the space of all available text such as sign languages (Yin et al., 2021), and non-digitized languages. Best practice recommendations identified by our work will not be directly applicable to this space. Additionally, a text-only focus precludes the need to develop methods to bridge and transfer information across multiple modalities like images, audio, video, etc., and our best practices might again not be applicable to those scenarios. In addition to modality, another macro dimension largely missing from our work is temporality. While domain distinctions might also capture some aspects of temporality,[2] none of our case studies focus on analyzing this dimension systematically. However, given the ubiquity of large pretrained language models and the difficulty and costs associated with re-training them, efficient and quick temporal adaptation is emerging as a crucial research question (Lazaridou et al., 2021). We leave exploration of both these missing dimensions to future work.

6.3 Broad Directions for Future Work

6.3.1 Looking Forward vs Looking Back

A major future direction to extend the work presented in this thesis will be assessing the predictive value of the retrospective conclusions derived from the case studies. Throughout our process of building up a set of best practices, our case studies have been *looking back*, i.e., they have been retrospective in nature. Though our experimental setups were motivated by pertinent results from prior studies, we did not explicitly *look forward*. In other words, we have not quantitatively measured the utility of our set of best practice recommendations in helping practitioners choose better models for new long tail dimensions. We leave such assessment to future case studies, which can accomplish this by evaluating on domains not tested in this work and incorporating hypothesis testing. Possible hypotheses may focus on measuring savings (in terms of GPU usage, data collection budget, model development time, etc.) arising from adopting our recommended transfer methods as a starting point, vs exploring a large space of methods. Another set of possible hypotheses focuses on recommended source-target distance measures predictively to choose transfer methods that would work best and measuring savings in comparison to exploring a larger method space.

6.3.2 Promising New Categories of Transfer Methods

A second crucial future direction is continually expanding the adaptation method taxonomy with promising new categories of transfer methods and evaluating their utility on the long tail. With rapidly increasing interest in the field of transfer learning, there are already several new categories

[2]For example, one of the domains we explore is literary text, which contains documents from books published before 1923.

that are strong contenders for inclusion in our taxonomy:

Retrieval-augmented methods: With advances in neural information retrieval, retrieval-augmented methods have started seeing a resurgence in the field of natural language processing. This modeling paradigm has been explored in the context of both improving pretrained language models (Guu et al., 2020), as well as improving downstream model performance (Lewis et al., 2020; Naik et al., 2021c). In the context of transfer learning, the retrieval-augmented paradigm could be exploited in two ways. The first way, which is primarily applicable in multi-source, continual, or incremental settings, consists of maintaining a knowledge store containing distributional information about domains seen so far. When encountering a new domain, information pertaining to relevant domains can be retrieved from the knowledge store and used to improve performance on the new domain. Prior work has accomplished this by designing an additional external "memory bank" to store distributional information about domains via an attention mechanism (Asghar et al., 2019), but there is scope to experiment with other representation mechanisms. The second way, which is applicable to any setting, consists of maintaining an external structured or unstructured source of facts (or knowledge), and retrieving pertinent information for a new domain or example. A toy example of such a setting could be the use of domain or language-specific gazetteers when adapting named entity recognition models. This could be an extremely useful setup for domains such as clinical/biomedical text, in which expert-curated knowledge sources like UMLS (Bodenreider, 2004) exist. In the context of our adaptation taxonomy, this method would again fit under the hybrid category since it involves changes to model architecture, as well as on-the-fly input representation (i.e. data) manipulation.

Prompting-based methods: With the resounding success of GPT-3 (Brown et al., 2020), especially in few-shot settings, interest in developing prompting-based methods for various NLP tasks has soared. Liu et al. (2021) provide a comprehensive survey of prompting methods and their use in NLP. These methods have opened up several new avenues for the field of transfer learning to explore. For example, prompting-based methods have been used as better pseudo-labelers (Schick and Schütze, 2021; Chintagunta et al., 2021), which is a strategy to leverage prompting to improve existing adaptation methods. A different avenue has tried to use prompting directly for adaptation by prompting models with a few exemplars from a new domain (Ben-David et al., 2021). Within our taxonomy, prompting-based adaptation methods would fit under the data-centric category since they primarily work by providing access to a subset of domain exemplars. Finally, recent work has been exploring the use of prompting for interesting meta tasks such as producing natural language descriptions of the distributional shift between domains (Zhong et al., 2022).

6.3.3 Standardizing Multi-Faceted Evaluation and Analysis

Finally, another interesting future direction is to make strides towards developing a standardized yet flexible protocol for follow-up quantitative and qualitative analyses, in addition to overall scores, to

support improved understanding of method performance. From our case studies, we recommend certain categories of analyses that we find useful, but there are still various choices that practitioners may end up making arbitrary selections for. For example, considering the category of quantitative performance analysis on different types of linguistic shifts, we quickly come up against several choices. The first choice lies in determining what set of categories of linguistic shifts would be comprehensive and sufficient. The second choice lies in determining how to operationalize every shift category chosen (e.g., evaluating on OOV tokens for lexical shift). The third choice lies in determining what degree of performance difference can be considered strongly indicative of a broader trend. An additional parameter at this step is determining sample sizes for the analysis. Therefore, developing a standardized protocol for analyses will primarily consist of two stages: (i) identifying and experimentally validating default settings for the set of possible choices and analyses, and (ii) developing a tool that allows practitioners to quickly set up default analyses, while also offering the flexibility to make different choices (for more seasoned practitioners). Prior work has attempted to build such standardized frameworks, though they were primarily focused on qualitative analyses only, with an eye towards scalability and reproducibility (Wu et al., 2019).

In addition to developing a standardized analysis protocol and framework, improving their utility by encouraging community adoption will also be a challenging and interesting problem to tackle. Following are some potential solutions that can be explored to promote adoption:

- Including optional questions about analyses conducted in the responsible NLP checklist, which is now a crucial part of a submission to the ACL Rolling Review. While this will not make analysis mandatory, it may make authors engage more critically with this topic.
- Organizing shared tasks focused on generating interesting insights from follow-up analyses of existing datasets and methods.
- Potentially making analysis sections mandatory in resource-focused venues (e.g., LREC, SemEval task description papers, resource and datasets tracks in NLP and ML conferences).

Implementing these solutions will be a complex undertaking and will need careful consideration and feedback from relevant stakeholders, but may help in making deeper performance analysis more mainstream.

6.4 Focusing on the Long Tail: Broader Impact

In closing, we would like to highlight that focusing on adapting to the long tail is valuable not just from an NLP perspective, but often also from the perspective of real-world impact and utility. At the macro-level, many crucial domains (e.g., clinical text) have been relegated to the long tail of NLP research, despite the fact that having strong adaptation techniques for these domains could enable rapid development of technologies with the potential to make strong social impact. Continuing our focus on language understanding, we describe two example high-impact scenarios: (i) assisting disability determination, and (ii) enabling rapid understanding of an increasing body of medical literature (e.g., literature on COVID-19).

In the United States, disability determination is the process by which the Social Security Administration (SSA) determines whether individuals are eligible to receive federal disability benefits, by virtue of being unable to pursue any gainful employment due to their medical condition(s). Individuals applying for these benefits must submit a case, which is then reviewed by case examiners (also called adjudicators), physicians and psychologists. During the review process, individuals may be asked to provide documentation describing what their medical condition is and how it limits their activities, when the condition began, and what tests and treatments have been pursued. Submitted documentation typically includes clinical records and accounts from the individual's doctors and hospitals. Once an initial acceptance or rejection decision is provided for a case, an individual may appeal it if desired. A single case can require an adjudicator to manually review hundreds of evidence pages to determine eligibility for benefits based on financial, medical, and functional criteria. Additionally, the program receives between 2 and 3 million new applications every year, and coupled with an aging workforce where larger numbers of adjudicators are expected to retire, continuing the solely manual review process looks to be increasingly infeasible (Desmet et al., 2020). NLP technologies are in a unique position to assist in improving throughput by automating certain sections of the process (e.g., terminology extraction, NER, document ranking, etc.) and making it easier for adjudicators to search for and locate relevant information quickly. However, the language used in clinical records submitted for adjudication is very different from sources used to train typical NER and ranking models (e.g., newswire, web text, etc.), and gathering annotation is extremely difficult due to data sharing restrictions and the need for domain experts. In this scenario, adaptation techniques can be used effectively to reduce in-domain annotation requirements and encourage model reuse over training new ones from scratch, and the resulting models, if accurate, can have massive social impact in a high-stakes setting.

The second scenario that we highlight is helping medical practitioners keep up with the massive and ever-increasing body of medical literature. As noted by Fiorini et al. (2017), medical literature is expanding at a rapid pace, with the PubMed Central repository of articles alone growing by more than 1000 articles per day. Keeping up with this large volume of literature is extremely difficult for medical practitioners with their already busy schedules. On the other hand, staying up-to-date with the latest advances in testing and treatment can help practitioners improve patient care and outcomes. This can be very crucial when dealing with novel diseases (or disease combinations), or emerging pandemic situations like the ongoing coronavirus (COVID-19) pandemic, in which very little is known beforehand and new treatments are constantly being developed and tested. Indeed since the start of the pandemic in December 2019, $\sim$245,000 articles on COVID-19 have been added to PubMed (Chen et al., 2021) and several datasets have been publicly released to encourage the development of techniques to mine valuable insights and evidence from this literature (Wang et al., 2020; Chen et al., 2020a, 2021). In such settings, there is massive scope for building NLP technologies to provide crucial assistance to practitioners such as identifying novel emerging concepts and their definitions/applications, identifying claims and supporting evidence from clinical trials, and identifying new relationships established between concepts by various studies (Hope

et al., 2021). Similar to the disability determination scenario, language in medical articles is different from sources used to build typical concept extraction, claim and evidence extraction, and relation extraction models, and gathering in-domain annotation is difficult since understanding this language requires a high level of domain expertise. Again, adaptation techniques can be used to build effective models for these tasks in data-scarce scenarios, and these models may provide crucial assistance to medical practitioners, having strong social impact.

Though we have only presented these two settings in detail, there are several such scenarios involving long-tail domains, in which developing effective NLP tools via adaptation can lead to much better real-world utility and higher social impact of NLP techniques. The existence of this potential for broader impact lends further motivation to the study of adaptation for the long tail in language understanding.

APPENDIX A

Meta-Analysis Coded Papers

Table A.1 provides an exhaustive list of model categories, both coarse and fine, tested in each study included in our meta-analysis. Some papers are surveys, position pieces, or meta-experiments in which case no method labels are assigned. Some papers use multiple methods, in which case we list all method labels, while some papers combine methods from different categories (indicated using '+').

Study	Coarse Method	Fine Method
Blitzer et al. (2007)	MC	FA
Howard and Ruder (2018)	DC	PT
Daumé III (2007)	MC	FA
Blitzer et al. (2006)	MC	FA
Conneau et al. (2017)	DC	PT
Jiang and Zhai (2007)	HY	IW
Liu et al. (2019a)	MC	LA
Søgaard and Goldberg (2016)	MC	LA
Cer et al. (2018)	DC+MC	PT+LA
Liu et al. (2015)	MC	LA
Eisenstein (2013)	–	–
Prettenhofer and Stein (2010)	MC	FA
Nguyen and Grishman (2015)	MC	FA
Mou et al. (2016)	–	–
Finkel and Manning (2009)	MC	FA
Li et al. (2012)	DC+MC+HY	LA+PL+IW
McClosky et al. (2010)	MC	EN

Zarrella and Marsh (2016)	DC	PT
Chiticariu et al. (2010)	MC	HE
Chan and Ng (2007)	DC+MC+HY	AL+IW+PI
Yang et al. (2017)	DC+MC	LA+PL+FP
Subramanya et al. (2010)	DC	PL
Plank and Moschitti (2013)	MC	FA
Rai et al. (2010)	DC+MC	AL+LA
Kim et al. (2017)	MC	LA
Romanov and Shivade (2018)	DC, MC	PT, FA
Jeong et al. (2009)	DC+HY	IW+PL
Tsuboi et al. (2008)	MC	LA
Huang et al. (2018)	MC	FA
Chan and Ng (2006)	MC	PI
Zhang et al. (2017)	MC	FA+LA
Szarvas et al. (2012)	MC	FA
Chen and Qian (2019)	MC	LA
Monroe et al. (2014)	MC	FA
Mohit et al. (2012)	DC+MC	LA+PL
Kim et al. (2016)	MC	FA
Alam et al. (2018)	MC	LA
Wang et al. (2019a)	–	–
Heilman and Madnani (2013)	MC	FA
Arnold et al. (2008)	MC	FA
Lin and Lu (2018)	MC	PA+FA
Yang and Eisenstein (2015)	MC	FA
Agirre and Lopez de Lacalle (2009)	MC	FA
Wang et al. (2018)	MC	LA
Braud and Denis (2014)	MC, HY	DS, PI, EN, FP
Agirre and Lopez de Lacalle (2008)	MC	FA
Duong et al. (2017)	DC, MC	PT, LA, PL
Pilán et al. (2016)	DC, MC, HY	FP, IW, DS, NO
Yu and Kübler (2011)	DC+HY	PL+IW
Tamkin et al. (2020)	–	–
Vu et al. (2020)	–	–
Ji et al. (2015)	MC+HY	AE+IW
Chen et al. (2020b)	MC	LA
Umansky-Pesin et al. (2010)	DC	PL
Lison et al. (2020)	DC	PL
Scheible and Schütze (2013)	DC+MC+HY	FP+PL+DS

Tan and Cheng (2009)	MC+HY	FP+IW
Gong et al. (2016)	MC	LA+PI
Sapkota et al. (2016)	MC	FA
Abdelwahab and Elmaghraby (2016)	DC	PT
Yin et al. (2015)	MC	FR
Wu et al. (2017)	DC+MC	AL+LA
Giménez-Pérez et al. (2017)	MC	FA
Nguyen et al. (2014)	DC+MC	PL+EN
Johnson et al. (2019)	MC	PI, FP
Tourille et al. (2017)	DC, MC, HY	FR, NO, DS
Plank et al. (2014)	HY	IW
Chen et al. (2018a)	MC	LA
Hangya et al. (2018)	DC+MC, MC	PT+FP, LA
Passonneau et al. (2014)	–	–
Chang et al. (2010)	–	–
Al Boni et al. (2015)	MC	PI
Huang et al. (2019)	DC+MC	PI+PL
Rodriguez et al. (2018)	MC	FP, PI
Wright and Augenstein (2020)	–	–
Li et al. (2019b)	DC, MC, HY	LA, FP, DS, PT
Gee and Wang (2018)	MC	PI
Vlad et al. (2019)	MC PI	
Yang et al. (2015)	MC	AE
Xing et al. (2018)	MC	LA
Yan et al. (2020)	MC	FA+LA
Lee et al. (2020)	DC	PL
Fares et al. (2018)	MC	LA, PI
Jochim and Schütze (2014)	MC, HY	AE, FP, IW, EN
Naik and Rose (2020)	MC	LA
Schröder and Biemann (2020)	–	–
Jiao et al. (2018)	DC	PT
Yang et al. (2018)	MC	PI
Li et al. (2019a)	MC	FA
Chalkidis et al. (2020)	DC	PT
Beryozkin et al. (2019)	DC	PL
Karunanayake et al. (2019)	MC	FA
Dhillon et al. (2012)	DC	PL
Dereli and Saraclar (2019)	DC+MC	FA+PT
Keung et al. (2020)	–	–

Aggarwal and Sadana (2019)	DC	PT
Huang and Lin (2016)	MC	PI
Wiedemann et al. (2019)	DC, MC	PT, LA
Kamath et al. (2019)	MC	LA
Akdemir (2020)	DC, MC	PT, LA

Table A.1: Adaptation method coding (both coarse and fine categories) for all papers included in our meta-analysis.

APPENDIX B

Coding Manual for Events

Task: Annotate entities and event triggers in clinical notes and doctor-patient conversations.

Format: Given a .txt file containing the tokenized clinical note or conversation transcript, label all text spans corresponding to entities and event triggers using the entity and event labels respectively, in the BRAT interface. Please do not edit the file otherwise, even if there are strange phrases resulting from the tokenization (eg: don 't). For now, we are not annotating any additional metadata such as event type, time or attributes.

Procedure: The annotation procedure is divided into two phases:

- **Phase 1:** Annotate all entity mentions in the conversation. This phase has been added in order to ensure that important information regarding event participants is not lost since the event annotation procedure only focuses on triggers.
- **Phase 2:** Annotate all event triggers in the text.

B.1 Phase 1: Entity Annotation

Our definition of entities encompasses two categories of terms:

- Named entities (usually, but not limited to, medications)
- Physical objects (usually, but not limited to, body parts and medications)

Note that we will be annotating all entities, irrespective of whether they participate in an event or not.

How to annotate entities:
Entities are always expressed as noun phrases. Following are some examples of entities (bolded text):

1. You went through all the **treatment medications**
2. I will write you a **script** for this test.
3. Please take that **chair**.
4. You should go to Kroger for **Midol**.
5. Put the **medicine** on your **hands**.
6. You should not take more than **five or six pills** daily.

While annotating entities, please keep the following points in mind:

- While annotating noun phrases, we will leave out determiners and pronouns, but include adjectives and quantifiers. Note that when we encounter phrases with mixed combinations of both (e.g., "all that good stuff"), we will give higher preference to discarding determiners and pronouns over keeping quantifiers (i.e., we will annotate "good stuff").
- If an entity is mentioned multiple times in the same utterance (e.g., due to disfluencies), we will annotate all occurrences. Example: Put the **cream**, uh yes, the **cream** on your hands daily.
- All annotated entities should be continuous phrases
- We will not annotate locations as entities. However, whenever phrases appear as location names but are being used to refer to groups of people working at that location, we will annotate the phrase. For example: "**Pathology** will be sending the results soon"

B.2 Phase 2: Event Annotation

Our basic definition of events draws from TimeBank and LitBank. Events are considered a cover term for situations that happen or occur. In other words, activities (dynamically unfolding processes), accomplishments (almost instantaneous occurrences) and achievements (occurrences which have some duration, but also a predetermined endpoint) are considered as events. Events may be punctual or last for a period of time. Predicates describing states or circumstances in which something obtains or holds true are also annotated as events. Common event types in our dataset include (but are not limited to) conditions, symptoms, tests, treatments, patient visits and changes in any of these events.

How to annotate events:
Events may be expressed as tensed or untensed verbs, nominalizations, adjectives, predicative clauses or prepositional phrases. Additionally, events might also occur as phrases (continuous or discontinuous). In our scheme, we will only label a single word in case of phrasal events (with some exceptions), based on certain guidelines. The following sections detail these guidelines and provide examples for annotating events from each of these syntactic categories.
Note: In each example, we have only marked events which exhibit the specific rule being described. Any additional events have been left unmarked.

B.2.1 Verb Events

Verb events are annotated as per the following rules:

1. Verb events can be tensed or untensed.
   (a) I **took** Midol for 6 months.
   (b) I want you to **take** Midol for the next few days.
2. For verb phrases we will only annotate the head of the phrase.
   (a) I have been **taking** Midol for 6 months.
3. For phrasal verbs, we will not mark the particles.
   (a) Maybe we can **go** off the Midol.
4. If the phrase contains an aspectual verb and a main verb, both will be annotated as separate events since they provide separate pieces of information.
   (a) I **started taking** Midol.
5. If the phrase contains aspectual and main verbs with the aspectual verb having auxiliary verbs, the auxiliary will be ignored.
   (a) You can **start taking** Midol.
6. When dealing with chains of verbs, all verbs which correspond to occurrences must be annotated except for modals.
   (a) You can **come** to **see** me if it gets worse.
   (b) I **went** to **see** my brother and it made my pain worse.
   (c) You have to **see** me next week.
7. We will not annotate any verb forms of “to be”, even when it is used as a main verb since it has no semantic content.

   (a) You must be uncomfortable (ignore "be" here)

8. **Phrasal Verbs:** This category is the only exception to our rule of annotating single word events. We will annotate phrasal verbs as spans of text instead of choosing a word to tag.

   (a) You should recover enough to be able to **get back** to work by next week.

B.2.2 Noun Events

Noun events are annotated as per the following rules:

1. Ignore determiners while annotating noun events.

   (a) You had the **surgery** last week?

   (b) How was the **scan**?

2. For noun phrase events, we only annotate the head of the phrase.

   (a) Your last **scan** came out healthy.

   (b) I see you had a heart **surgery** in 2002.

3. For noun phrase events with a light predicate, both elements are tagged as events since they provide different aspects of event information.

   (a) Did you **get** the pap **smear** I recommended?

4. For cases where both nouns and verbs refer to the same event, we will mark both separately, keeping in mind that they refer to the same event.

   (a) Your **surgery** was **done** last year, was it?

5. **Compound nouns:** Unlike phrasal verbs, we will not annotate non-hyphenated compound nouns as spans of text. The reason for this is that deciding whether a non-hyphenated noun phrase is a compound noun is not as trivial as detecting phrasal verbs and often uses inherent domain knowledge of whether this phrase is more commonly used as a compound rather than independently using the nouns it is composed of.

   (a) Let me check your blood **pressure**.

B.2.3 Predicative Clauses

For predicative clauses, only the predicative element is tagged. If the predicative element has a head, we tag the head. If not, we tag the entire element.

1. Midol will be **good** for this.

2. You seem to be **on** Midol.

B.2.4 Prepositional Phrases

For prepositional phrases, we use a similar strategy as predicative clauses. However, in some cases nouns inside the prepositional phrase can also be eventive, referring to a parent/ sub event of the event expressed by the prepositional phrase. In these scenarios, we also annotate the noun as an event (refer to example 2).

1. Did you continue to experience cramps when **on** Midol?
2. Did you continue to experience cramps when **on** your **periods**?

B.2.5 Adjective Events

State events (or changes in states) are often expressed as predicative adjectives or as adjectives in light predicate constructions.

1. The arrhythmia is **worse**.
2. My cold is **better** now.
3. This treatment has been **good**.
4. Your leg got a little more **swollen**.

B.2.6 Causative Predicates

Causative predicates have their own set of rules to decide which of the multiple sub-events involved in these scenarios should be annotated. Causative predicatives usually fall into one of the following cases:

1. EXPR <causal_verb> EXPR (explicit expressions like “this medicine caused my blood pressure to go up”)
2. EXPR <discourse_marker> EXPR (implicit expressions like “I took Midol and felt better”)

In both cases, if the expressions are events, they are tagged (entities are ignored). If the causal expression is explicit, the causal predicate is also marked as an event. Examples:

1. This medicine **caused** my blood pressure to **go** up
2. I **took** Midol and **felt** better

B.2.7 Excluded Event Types

In our event annotation, we exclude the following event types:

1. Generic events should not be tagged. For example:
   (a) This medicine works well for cramps. (Both "works" and "cramps" will not be tagged here)
2. Subordinating verbs whose complements are generic events should not be tagged. For example:
   (a) I talked about the use of Midol for cramps (Here "talked" would not be annotated as an event)

B.2.8 Interesting Cases

These cases are interesting event expression patterns in medical conversational data which do not seem to be clearly dealt with under TimeBank guidelines. Some of these cases have been identified during repeated rounds of annotation.

1. **Events asked about in questions:** These events can be divided into two types:
   (a) Events which actually occurred, which should be marked since they have occurred. For example: What have you **applied** for?
   (b) Events which have not occurred. These events should also be marked because they have been queried about under the expectation that there was a chance of them occurring. As such, they should be considered as hypothetical events and marked. For example: Are you **following** the instructions given to you?
2. **Events occurring in commands, suggestions or requests:** Commands, suggestions or requests are very infrequent in narratives, but a lot more frequent in doctor-patient conversations. It is valuable to mark events in suggestions and requests because a lot of them are treatment-related and likely to be important for downstream tasks, especially SOAP note generation. These events are usually hypothetical in nature. For example:
   (a) Well you've got to **start** some other medicine.
   (b) I can **send** you to **see** a pediatrician.

   This category of events also includes events mentioned when doctors are "thinking out loud" and debating different treatment plans with the patient (a very common occurrence in many of the conversations). For example:
   (a) I can **write** you a prescription, and we can **check** your **pressure** everyday.

(b) You can **continue** with this medication for now, but we will **stop** if it **leads** to side **effects**.

3. **Directive events:** In many conversations, doctors often issue directives to patients to guide the conversation. Though these directives are actions (which are included in our definition of events), they do not seem particularly relevant. So, we will not annotate directive events. For example:

   (a) Take a seat.

   (b) Let's talk about your pain.

   In both these examples, "take" and "talk" will not be annotated as events.

4. **Stative events:** Timebank guidelines are very restrictive with respect to stative events, only allowing for annotations in state changes. But as mentioned earlier, we will annotate all stative events, irrespective of whether there are any changes in state during the course of the conversation. Annotating stative events seems necessary since they seem to be especially important for chronic condition cases (eg: diabetes, hypertension). For example:

   (a) You have been suffering from **cardiomyopathy** for a while now.

5. **Activity patterns:** TimeBank does not tag events which represent clear patterns of activity (eg:She takes Midol regularly). But we will be tagging those since they are important markers for key patient activities such as taking medication, checking symptoms etc. For example:

   (a) Do you **take** your blood **pressure** daily?

6. **Entity-Event precedence:** If the term you are looking at is a phrase that could reasonably be annotated as either an entity or an event and the actual label to be assigned depends on deducing more information from the entire context, we will always assign the "event" label. A good example here is the phrase "shots". It can occur as an entity (eg: You can stop taking the B12 shots) or as an event (eg: Have you been keeping up with your scheduled B12 shots?). Sometimes it is possible to make out whether these phrases have been used as entities or events, but they are often used in generic contexts where it is not feasible to make this distinction (eg: side effects). Therefore, we will always label these as events.

APPENDIX C

Dataset Examples

C.1 Existing Datasets Used in this Thesis

C.1.1 CoNLL 2003 Named Entity Recognition Dataset

Entity types present in this dataset: persons, organizations, locations, and miscellaneous. Following are some example annotated sentences from this dataset:

- EU rejects German call to boycott British lamb
- Only France and Britain backed Fischler's proposal
- Germany imported 47,600 sheep from Britain last year, nearly half of total imports
- Rare Hendrix song draft sells for almost $ 17,000
- German July car registrations up 14.2 pct yr / yr
- Volkswagen AG won 77,719 registrations, slightly more than a quarter of the total
- – Dimitris Kontogiannis, Athens Newsroom +301 3311812-4
- Bayer sets $ 100 million six-year bond
- Port conditions update - Syria - Lloyds Shipping
- Polish diplomat denies nurses stranded in Libya

C.1.2 i2b2 2006 Protected Health Information Identification Dataset

Entity types present in this dataset (after mapping): persons, organizations, locations, and miscellaneous. Following are some example annotated sentences from this dataset:

- Mr. Blind is a 79-year-old white white male with a history of diabetes mellitus, inferior myocardial infarction, who underwent open repair of his increased diverticulum November 13th at Sephsandpot Center
- Lives in Merca
- His chest was clear, and therefore on 11/03/02 the patient was discharged with a followup plan
- He was transfused one unit of platelets prior to discharge with followup at the DFCI on 11/06 with a transfusion unit and on 11/09 with Dr. Charla B Titchekote for additional bone marrow biopsy .
- TICE D. FOUTCHJESC , M.D. UU2 FK795/004653
- On 03/24/98 , an echocardiogram revealed a pericardial effusion
- The patient lives in Jer by herself
- She has sustained a right inter-trochanteric hip fracture both treated at Hoseocon Medical Center and transferred to the Heaonboburg Linpack Grant Medical Center for further care by Dr. Stable
- XRT, Friday, 10am 05/22/02 scheduled, Dr. Xellcaugh, call next week, No Known Allergies
- If you need additional information please call 605-304-8547

C.1.3 i2b2 2014 Protected Health Information Identification Dataset

Entity types present in this dataset (after mapping): persons, locations, and miscellaneous. Following are some example annotated sentences from this dataset:

- I think Dr. Gipson is basically doing very well.
- He was recently in Italy last week with very difficult events concerning his father's ill health
- A/P: 75yr man s/p sig colostomy for diversion, s/p IR drainage of postop collection, now with FTT, acute renal failure with FENA>1% and possible UTI, non-gap metabolic acidodis
- Please contact BCH Surgical Service with any questions
- Earl N. Morrow, M.D.
- 1666 Keats Street GLENN, OLIVIA
- eScription document:2-5091452 EMSSten Tel
- DV: 05/25/79
- Union City, CT 33636
- Your patient Lauren Ferrara came into the office today for a follow-up visit

C.1.4 i2b2 2010 Medical Concept Extraction Dataset

Entity types present in this dataset: problems, tests, and treatments.[1] Following are some example annotated sentences from this dataset:

[1]Note that for all experiments using this dataset we only identify entities without performing additional type identification.

- The patient had undergone treatment with interferon and Ribavirin
- The patient was admitted with concern for hepatorenal syndrome
- Over the following 5 days, the patient's creatinine improved marginally to 2.7
- 4. Vancomycin 1 gm IV bid
- 3. Nephrolithiasis
- The patient was thrombocytopenic with a platelet count of 49 on the 23
- The patient was continued on vancomycin therapy for his previously diagnosed Methicillin resistant, coagulase negative Staph bacteremia
- 2015-10-27 Open reduction and internal fixation of right tib/fib fractures
- Attention deficit disorder (diagnosed @ 14 years of age)
- GLUCOSE - 101 LACTATE - 2.9 * NA+ - 142 K+ - 3.6 CL- - 102 TCO2 - 26

C.1.5 TimeBank Event Extraction Dataset

Following are some example annotated sentences from this dataset (note that tokens highlighted in yellow are events):

- The thrift holding company said it expects to obtain regulatory approval and complete the transaction by year-end
- The remaining $ 40 million can be used over three years for oil and gas acquisitions, the company said
- The company put up "virtually all" of its oil and gas properties as collateral, he said
- The Bureau of Labor Statistics said the economy added 350,000 jobs last month, far above the 235,000 forecast by economists
- The demand for workers also led employers to raise wages last month
- But economists said the wage increase was not enough to raise any concerns about higher inflation
- The surge in jobs reflects a remarkable confluence of positive and self-reinforcing economic forces
- Spontaneous applause echoed through the chamber and public galleries as the crucial vote passed by a wide margin
- Treasurer Peter Costello, Environment Minister Robert Hill and Attorney General Daryl Williams all voted to support the republic Friday
- Monarchists hope to defeat the republic at the referendum

C.1.6 LitBank Literary Event Extraction Dataset

Following are some example annotated sentences from this dataset (note that tokens highlighted in yellow are events):

- Certain it is that, some fifteen or twenty years after the settlement of the town, the wooden jail was already marked with weather-stains and other indications of age, which gave a yet darker aspect to its beetle-browed and gloomy front
- "Of course, we'll take over your furniture, mother," Winnie had remarked
- His work was in a way political, he told Winnie once
- She would have, he warned her, to be very nice to his political friends
- The mean aspect of the shop surprised her
- The change from the Belgravian square to the narrow street in Soho affected her legs adversely
- An awful panic spread through the whole building
- But his father's friend, of course, dismissed him summarily as likely to ruin his business
- It stifles me
- I can detect the scent through all the foul smells lounging in the air

C.1.7 i2b2 2012 Medical Event Extraction Dataset

Following are some example annotated sentences from this dataset (note that tokens highlighted in yellow are events):

- Later that am, she stood from her wheelchair, had a prodrome of LH and then reports LOC and fall
- She reports that her BP was read at 60/48 after her fall
- She also denies any recent cough, chest pain, chest palpitations, heart racing
- Mr. Williams is an 85 yo gentleman who has a known cardiac history and has had a h/o worsening chest pain and shortness of breath
- He underwent cardiac catheterization which showed an 80% LAD lesion, chronically occluded RCA, anneurysmal mid LCX w/50% lesion
- He was started on heparin and coumadin for anticoagulation on POD#5
- She was found to have widely metastatic ovarian carcinoma
- The patient was transferred to the Fairm of Ijordcompmac Hospital
- She had previous history of claudication
- There was an occlusion in the proximal calf, of the peroneal and anterior tibial arteries

C.1.8 TimeBank-Dense Temporal Ordering Dataset

Classes present in this dataset: after, before, includes, is included, simultaneous, and vague. Following are some annotated examples from this dataset (note that the annotated temporal relationship holds between the events highlighted in yellow):

- Har-Shefi said she heard Amir talk about killing Rabin but did not tell the police because she did not believe he was serious
  Label: (talk, serious) -> vague
- Har-Shefi acknowledged she told police interrogators that Rabin was a traitor and that she prayed for him to have a heart attack and die
  Label: (die, attack) -> vague
- Amir, 27, is serving a life sentence for the November 1995 assassination of Rabin at a Tel Aviv peace rally
  Label: (rally, assassination) -> includes
- Amir, 27, is serving a life sentence for the November 1995 assassination of Rabin at a Tel Aviv peace rally
  Label: (assassination, serving) -> before
- The major harm from Asia is likely to come from the plunge in the value of many Asian currencies relative to the dollar, a situation that is expected to lead to a surge of inexpensive imports into the United States, hurting American competitors
  Label: (surge, hurting) -> vague
- WASHINGTON _ The economy createdjobs at a surprisingly robust pace in January, the government reported on Friday, evidence that America's economic stamina has withstood any disruptions caused so far by the financial tumult in Asia
  Label: (reported, withstood) -> after
- The gain left wages 3.8 percent higher than a year earlier, extending a trend that has given back to workers some of the earning power they lost to inflation in the last decade
  Label: (left, trend) -> is included
- I think it's excellent for the company. But investors are approaching the changes with caution shares of AT and T down nearly four at sixty-one and a half.
  Label: (changes, excellent) -> simultaneous
- The changes are part of a one point six billion dollar cost cutting initiative to revitalize its position in the telecommunications business.
  Label: (revitalize, cutting) -> vague
- The changes are part of a one point six billion dollar cost cutting initiative to revitalize its position in the telecommunications business.
  Label: (initiative, cutting) -> vague

C.1.9 MultiNLI Natural Language Inference Dataset

Classes present in this dataset: entailment, contradiction, and neutral. Following are some annotated examples from this dataset:

- **Premise:** The Old One always comforted Ca'daan, except today.
  Hypothesis: Ca'daan knew the Old One very well.
  Label: Neutral
- **Premise:** Your gift is appreciated by each and every student who will benefit from your generosity.
  Hypothesis: Hundreds of students will benefit from your generosity.
  Label: Neutral
- **Premise:** yes now you know if if everybody like in August when everybody's on vacation or something we can dress a little more casual or
  Hypothesis: August is a black out month for vacations in the company.
  Label: Contradiction
- **Premise:** At the other end of Pennsylvania Avenue, people began to line up for a White House tour.
  Hypothesis: People formed a line at the end of Pennsylvania Avenue.
  Label: Entailment
- **Premise:** Met my first girlfriend that way.
  Hypothesis: I didn't meet my first girlfriend until later.
  Label: Contradiction
- **Premise:** 8 million in relief in the form of emergency housing.
  Hypothesis: The 8 million dollars for emergency housing was still not enough to solve the problem.
  Label: Neutral
- **Premise:** Now, as children tend their gardens, they have a new appreciation of their relationship to the land, their cultural heritage, and their community.
  Hypothesis: All of the children love working in their gardens.
  Label: Neutral
- **Premise:** At 8:34, the Boston Center controller received a third transmission from American 11
  Hypothesis: The Boston Center controller got a third transmission from American 11.
  Label: Entailment
- **Premise:** I am a lacto-vegetarian.
  Hypothesis: I enjoy eating cheese too much to abstain from dairy.
  Label: Neutral
- **Premise:** someone else noticed it and i said well i guess that's true and it was somewhat melodious in other words it wasn't just you know it was really funny
  Hypothesis: No one noticed and it wasn't funny at all.
  Label: Contradiction

C.2 New Datasets Contributed by this Thesis

C.2.1 MTSamples Medical Event Extraction Dataset

Following are some example annotated sentences from this dataset (note that tokens highlighted in yellow are events):

- Closure complex, open wound
- Bilateral explantation and removal of ruptured silicone gel implants
- She had no prior history of skin cancer
- She has noted progressive hardening and distortion of the implant
- The patient desires a repeat section
- ESTIMATED BLOOD LOSS: 800 mL
- Complications: None
- The fascia was incised in the midline and extended laterally using Mayo scissors
- The pulmonary arterial pressures were noted to be 31/14/21 mmHg
- Following this, the catheter was exchanged over the guidewire for 6-French JR4 diagnostic catheter

C.2.2 TDDiscourse Temporal Ordering Dataset

Classes present in this dataset: after, before, includes, is included, and simultaneous. Following are some annotated examples from this dataset (note that event tokens involved in various temporal relations are highlighted in yellow):

- JERUSALEM (AP) _ Taking the stand in her own defense, a friend of Yitzhak Rabin's assassin said Friday that she regretted calling the prime minister a traitor and praying for his death. Margalit Har-Shefi, 22, has pleaded innocent to charges that she failed to report Yigal Amir's plan to kill Rabin. She took the stand for more than four hours Friday in a Tel Aviv magistrate's court. Amir, 27, is serving a life sentence for the November 1995 assassination of Rabin at a Tel Aviv peace rally. Newspaper reports have said Amir was infatuated with Har-Shefi and may have been trying to impress her by killing the prime minister.
  Label: (taking, took) -> simultaneous
  Label: (taking, serving) -> is included
  Label: (taking, assassination) -> after
  Label: (taking, rally) -> after
  Label: (taking, said) -> after
- CANBERRA, Australia (AP) _ Qantas will almost double its flights between Australia and India by August in the search for new markets untouched by the crippling Asian financial crisis. This move comes barely a month after Qantas suspended a number of services between Australia,

Indonesia, Thailand and Malaysia in the wake of the Asian economic crisis. The airline has also cut all flights to South Korea. Qantas plans daily flights between Sydney and Bombay, up from the current four flights a week, to boost business and tourism ties with India, the airline announced Friday. In a joint statement with Tourism Minister Andrew Thomson, it said two new flights would leave Bombay on Monday and Tuesday nights from March 30, with the third departing each Thursday from August 6. This will add nearly 700 seats a week on the route. Thomson, in India to talk to tourism leaders, said the flights would provide extra support to the growing tourism market.
Label: (double, plans) -> is included
Label: (double, boost) -> before
Label: (double, leave) -> is included
Label: (double, talk) -> after
Label: (double, provide) -> is included

C.2.3 Stress Tests Natural Language Inference Test Set

Classes present in this dataset: entailment, contradiction, and neutral. Following are some annotated examples from this dataset:

- **Premise:** As a result, EPA could not ensure that it was directing its efforts toward the environmental problems that were of greatest concern to citizens or posed the greatest risk to the health of the population or the environment itself.
  Hypothesis: As a result, EPA could not ensure that it was directing its efforts toward the environmental problems that were of greatest concern to noncitizen or posed the greatest risk to the health of the population or the environment itself.
  Label: Contradiction
- **Premise:** Because several passengers on United 93 described three hijackers on the plane, not four, some have wondered whether one of the hijackers had been able to use the cockpit jump seat from the outset of the flight.
  Hypothesis: Because several passengers on United 93 described three hijackers on the plane, not four, some have wondered whether one of the hijackers had been able to use the cockpit jump seat from the end of the flight.
  Label: Contradiction
- **Premise:** Renu can do a piece of work in 8 days, but with the help of her friend Suma, she can do it in 4 days.
  Hypothesis: Renu can do a piece of work in more than 4 days, but with the help of her friend Suma, she can do it in 4 days.
  Label: Entailment
- **Premise:** Renu can do a piece of work in more than 4 days, but with the help of her friend Suma, she can do it in 4 days.

Hypothesis: Renu can do a piece of work in 8 days, but with the help of her friend Suma, she can do it in 4 days.
Label: Neutral

- **Premise:** Renu can do a piece of work in 8 days, but with the help of her friend Suma, she can do it in 4 days.
  Hypothesis: Renu can do a piece of work in 2 days, but with the help of her friend Suma, she can do it in 4 days.
  Label: Contradiction
- **Premise:** because like Tech is known to be a good engineering school and A and M maybe is known more for computers
  Hypothesis: Tech is known as a good place for engineering, but I think that it is overrated and false is not true.
  Label: Neutral
- **Premise:** The levadas were largely built by slave laborers from Africa, whose primary employment was on sugar plantations and true is true and true is true and true is true and true is true and true is true.
  Hypothesis: The levadas were built by 10,000 slaves.
  Label: Neutral
- **Premise:** Possibly no other country has had such a turbulent history.
  Hypothesis: The country's history has been turbulent and true is true.
  Label: Entailment
- **Premise:** As a result, EPA could not ensure that it was directing its efforts toward the environmental problems that were of greatest concern to citizens or posed the greatest risk to the health of the population or the environment itself.
  Hypothesis: EPA couldn't ensure it was directing its efofrts toward the environmental problem.
  Label: Entailment
- **Premise:** But if you do, kill them.
  Hypothesis: If the situation is that, you should kikl them.
  Label: Entailment

C.2.4 EQUATE Natural Language Inference Test Set

Classes present in this dataset: entailment, contradiction, and neutral. Following are some annotated examples from this dataset:

- **Premise:** Sam had 9.0 dimes in his bank and his dad gave him 7.0 dimes.
  Hypothesis: Sam has 16.0 dimes now.
  Label: Entailment

- **Premise:** Sam had 9.0 dimes in his bank and his dad gave him 7.0 dimes.
  Hypothesis: Sam has 17.0 dimes now.
  Label: Contradiction
- **Premise:** Lepore said he was moved to photograph the slumbering sentries after witnessing the same guard napping on three occasions.
  Hypothesis: Joey Lepore says he took photos of one guard sleeping at post three times.
  Label: Entailment
- **Premise:** It will mark the first time four women have been in space at one time.
  Hypothesis: Four women are aboard same spacecraft for first time.
  Label: Neutral
- **Premise:** In 1956 Accardo won the Geneva Competition and in 1958 became the first prize winner of the Paganini Competition in Genoa. He has recorded Paganini's famous 24 Caprices (re-recorded in 1999) for solo violin and was the first to record all six of the Paganini Violin Concertos.
  Hypothesis: Accardo composed 24 Caprices.
  Label: Neutral
- **Premise:** During Reinsdorf's 24 seasons as chairman of the White Sox, the team has captured American League division championships three times, including an AL Central title in 2000.
  Hypothesis: The White Sox have won 24 championships.
  Label: Entailment
- **Premise:** stocks nifty future call today: Sensex Weak and Nifty flat, Today best stock trading call on 3 Sept, Free nifty future stock tips, BHEL , Tata motor gain
  Hypothesis: Sensex and Nifty up, 2 sept Nifty stock market trading tips and top nifty gainers and losers on Monday, Indian stock market tips today stocks nifty future call today
  Label: Contradiction
- **Premise:** SENSEX Nifty up, Today stocks nifty future trading tips and call on Thursday 22 Aug, Nifty top gainers and losers stocks $\sim$ stocks nifty future call today
  Hypothesis: Sensex down 74.58 points, Nifty future tips, Tomorrow nifty future trading call on Wednesday 21 Aug, Nifty gainers and losers $\sim$ stocks nifty future call today
  Label: Contradiction
- **Premise:** Tim has 350 pounds of cement in 100, 50, and 25 pound bags.
  Hypothesis: Tim has less than 750 pounds of cement in 100, 50, and 25 pound bags.
  Label: Contradiction
- **Premise:** Mr Yadav spends 60% of his monthly salary on consumable items and 50% of the remaining on clothes and transport.
  Hypothesis: Mr Yadav spends 10% of his monthly salary on consumable items and 50% of the remaining on clothes and transport.
  Label: Contradiction

Bibliography

O. Abdelwahab and A. Elmaghraby. UofL at SemEval-2016 task 4: Multi domain word2vec for Twitter sentiment classification. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 164–170, San Diego, California, June 2016. Association for Computational Linguistics. doi: 10.18653/v1/S16-1024. URL https://www.aclweb.org/anthology/S16-1024.

N. Abe and H. Mamitsuka. Query learning strategies using boosting and bagging. In *Proceedings of the Fifteenth International Conference on Machine Learning*, pages 1–9, 1998.

K. Aggarwal and A. Sadana. NSIT@NLP4IF-2019: Propaganda detection from news articles using transfer learning. In *Proceedings of the Second Workshop on Natural Language Processing for Internet Freedom: Censorship, Disinformation, and Propaganda*, pages 143–147, Hong Kong, China, Nov. 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-5021. URL https://www.aclweb.org/anthology/D19-5021.

E. Agirre and O. Lopez de Lacalle. On robustness and domain adaptation using SVD for word sense disambiguation. In *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*, pages 17–24, Manchester, UK, Aug. 2008. Coling 2008 Organizing Committee. URL https://www.aclweb.org/anthology/C08-1003.

E. Agirre and O. Lopez de Lacalle. Supervised domain adaption for WSD. In *Proceedings of the 12th Conference of the European Chapter of the ACL (EACL 2009)*, pages 42–50, Athens, Greece, Mar. 2009. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/E09-1006.

R. Aharoni and Y. Goldberg. Unsupervised domain clusters in pretrained language models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7747–7763, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.692. URL https://aclanthology.org/2020.acl-main.692.

A. Akdemir. Research on task discovery for transfer learning in deep neural networks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: Student*

Research Workshop, pages 33–41, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-srw.6. URL https://www.aclweb.org/anthology/2020.acl-srw.6.

M. Al Boni, K. Zhou, H. Wang, and M. S. Gerber. Model adaptation for personalized opinion analysis. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 769–774, Beijing, China, July 2015. Association for Computational Linguistics. doi: 10.3115/v1/P15-2126. URL https://www.aclweb.org/anthology/P15-2126.

F. Alam, S. Joty, and M. Imran. Domain adaptation with adversarial training and graph embeddings. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1077–1087, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-1099. URL https://www.aclweb.org/anthology/P18-1099.

J. F. Allen. Towards a general theory of action and time. *Artificial intelligence*, 23(2):123–154, 1984.

J. F. Allen. Time and time again: the many ways to represent time. *International Journal of Intelligent System*, 6(4):341–355, 1991.

J. F. Allen and P. J. Hayes. A common-sense theory of time. *IJCAI*, 85:528–531, 1985.

E. Alsentzer, J. Murphy, W. Boag, W.-H. Weng, D. Jindi, T. Naumann, and M. McDermott. Publicly available clinical BERT embeddings. In *Proceedings of the 2nd Clinical Natural Language Processing Workshop*, pages 72–78, Minneapolis, Minnesota, USA, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/W19-1909. URL https://www.aclweb.org/anthology/W19-1909.

V. Ambati. Active learning and crowdsourcing for machine translation in low resource scenarios. *2011*, 2011.

G. Angeli and C. D. Manning. NaturalLI: Natural logic inference for common sense reasoning. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 534–545, Doha, Qatar, Oct. 2014. Association for Computational Linguistics. doi: 10.3115/v1/D14-1059. URL https://www.aclweb.org/anthology/D14-1059.

J. Araki and T. Mitamura. Open-domain event detection using distant supervision. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 878–891, Santa Fe,

New Mexico, USA, Aug. 2018. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/C18-1075.

A. Arnold, R. Nallapati, and W. W. Cohen. Exploiting feature hierarchy for transfer learning in named entity recognition. In *Proceedings of ACL-08: HLT*, pages 245–253, Columbus, Ohio, June 2008. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/P08-1029.

N. Asghar, L. Mou, K. A. Selby, K. D. Pantasdo, P. Poupart, and X. Jiang. Progressive memory banks for incremental domain adaptation. In *International Conference on Learning Representations*, 2019.

A. Axelrod, X. He, and J. Gao. Domain adaptation via pseudo in-domain data selection. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 355–362, Edinburgh, Scotland, UK., July 2011. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/D11-1033.

E. Bach. The algebra of events. *Linguistics and philosophy*, pages 5–16, 1986.

F. Bai, A. Ritter, and W. Xu. Pre-train or annotate? domain adaptation with a constrained budget. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5002–5015, Online and Punta Cana, Dominican Republic, Nov. 2021. Association for Computational Linguistics. URL https://aclanthology.org/2021.emnlp-main.409.

J. Balazs, E. Marrese-Taylor, P. Loyola, and Y. Matsuo. Refining raw sentence representations for textual entailment recognition via attention. In *Proceedings of the 2nd Workshop on Evaluating Vector Space Representations for NLP*, pages 51–55, Copenhagen, Denmark, Sept. 2017. Association for Computational Linguistics. doi: 10.18653/v1/W17-5310. URL https://www.aclweb.org/anthology/W17-5310.

J. Barwise and R. Cooper. Generalized quantifiers and natural language. In *Philosophy, Language, and Artificial Intelligence*, pages 241–301. Springer, 1981.

R. Bawden, R. Sennrich, A. Birch, and B. Haddow. Evaluating discourse phenomena in neural machine translation. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1304–1313, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-1118. URL https://www.aclweb.org/anthology/N18-1118.

B. Beizer. *Software Testing Techniques*. Dreamtech Press, 2003.

Y. Belinkov and Y. Bisk. Synthetic and natural noise both break neural machine translation. In *International Conference on Learning Representations*, 2018.

E. Ben-David, N. Oved, and R. Reichart. Pada: A prompt-based autoregressive approach for adaptation to unseen domains. *arXiv preprint arXiv:2102.12206*, 2021.

S. Ben-David, J. Blitzer, K. Crammer, A. Kulesza, F. Pereira, and J. W. Vaughan. A theory of learning from different domains. *Machine learning*, 79(1):151–175, 2010.

E. M. Bender. On achieving and evaluating language-independence in nlp. *Linguistic Issues in Language Technology*, 6(3):1–26, 2011.

L. Bentivogli, E. Cabrio, I. Dagan, D. Giampiccolo, M. L. Leggio, and B. Magnini. Building textual entailment specialized data sets: a methodology for isolating linguistic phenomena relevant to inference. In *Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC'10)*, Valletta, Malta, May 2010. European Languages Resources Association (ELRA). URL http://www.lrec-conf.org/proceedings/lrec2010/pdf/478_Paper.pdf.

J. Berant, V. Srikumar, P.-C. Chen, A. Vander Linden, B. Harding, B. Huang, P. Clark, and C. D. Manning. Modeling biological processes for reading comprehension. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1499–1510, Doha, Qatar, Oct. 2014. Association for Computational Linguistics. doi: 10.3115/v1/D14-1159. URL https://www.aclweb.org/anthology/D14-1159.

G. Beryozkin, Y. Drori, O. Gilon, T. Hartman, and I. Szpektor. A joint named-entity recognizer for heterogeneous tag-sets using a tag hierarchy. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 140–150, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1014. URL https://www.aclweb.org/anthology/P19-1014.

S. Bethard. ClearTK-TimeML: A minimalist approach to TempEval 2013. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 10–14, Atlanta, Georgia, USA, June 2013a. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/S13-2002.

S. Bethard. A synchronous context free grammar for time normalization. In *Conference on Empirical Methods in Natural Language Processing*, page 821. NIH Public Access, Oct. 2013b. URL https://www.aclweb.org/anthology/D13-1078.pdf.

S. Bethard, L. Derczynski, G. Savova, J. Pustejovsky, and M. Verhagen. SemEval-2015 task 6: Clinical TempEval. In *Proceedings of the 9th International Workshop on Semantic Eval-*

uation (SemEval 2015), pages 806–814, Denver, Colorado, June 2015. Association for Computational Linguistics. doi: 10.18653/v1/S15-2136. URL https://www.aclweb.org/anthology/S15-2136.

S. Bethard, G. Savova, W.-T. Chen, L. Derczynski, J. Pustejovsky, and M. Verhagen. SemEval-2016 task 12: Clinical TempEval. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 1052–1062, San Diego, California, June 2016. Association for Computational Linguistics. doi: 10.18653/v1/S16-1165. URL https://www.aclweb.org/anthology/S16-1165.

S. Bethard, G. Savova, M. Palmer, and J. Pustejovsky. SemEval-2017 task 12: Clinical TempEval. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 565–572, Vancouver, Canada, Aug. 2017. Association for Computational Linguistics. doi: 10.18653/v1/S17-2093. URL https://www.aclweb.org/anthology/S17-2093.

D. Biber. *Variation across speech and writing*. Cambridge University Press, 1991.

D. Biber and S. Conrad. *Register, Genre, and Style*. Cambridge Textbooks in Linguistics. Cambridge University Press, 2009. doi: 10.1017/CBO9780511814358.

J. Blitzer, R. McDonald, and F. Pereira. Domain adaptation with structural correspondence learning. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, pages 120–128, Sydney, Australia, July 2006. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/W06-1615.

J. Blitzer, M. Dredze, and F. Pereira. Biographies, Bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 440–447, Prague, Czech Republic, June 2007. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/P07-1056.

S. L. Blodgett, S. Barocas, H. Daumé III, and H. Wallach. Language (technology) is power: A critical survey of "bias" in NLP. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5454–5476, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.485. URL https://aclanthology.org/2020.acl-main.485.

D. G. Bobrow. Natural language input for a computer problem solving system. 1964.

O. Bodenreider. The unified medical language system (umls): Integrating biomedical terminology, 2004.

J. Bos and K. Markert. Recognising textual entailment with logical inference. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pages 628–635, Vancouver, British Columbia, Canada, Oct. 2005. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/H05-1079.

S. R. Bowman, G. Angeli, C. Potts, and C. D. Manning. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642, Lisbon, Portugal, Sept. 2015. Association for Computational Linguistics. doi: 10.18653/v1/D15-1075. URL https://www.aclweb.org/anthology/D15-1075.

P. Bramsen, P. Deshpande, Y. K. Lee, and R. Barzilay. Inducing temporal graphs. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, pages 189–198, Sydney, Australia, July 2006. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/W06-1623.

J. Brandtler. On aristotle and baldness: Topic, reference, presupposition of existence, and negation. *Working papers in Scandinavian syntax*, 77:177–204, 2006.

C. Braud and P. Denis. Combining natural and artificial examples to improve implicit discourse relation identification. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 1694–1705, Dublin, Ireland, Aug. 2014. Dublin City University and Association for Computational Linguistics. URL https://www.aclweb.org/anthology/C14-1160.

L. Breitfeller, A. Naik, and C. Rose. Stage: Tool for automated extraction of semantic time cues to enrich neural temporal ordering models. *arXiv preprint arXiv:2105.07314*, 2021.

T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, et al. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*, 2020.

S. Buchholz and E. Marsi. CoNLL-x shared task on multilingual dependency parsing. In *Proceedings of the Tenth Conference on Computational Natural Language Learning (CoNLL-X)*, pages 149–164, New York City, June 2006. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/W06-2920.

P. Budzianowski, T.-H. Wen, B.-H. Tseng, I. Casanueva, S. Ultes, O. Ramadan, and M. Gašić. MultiWOZ - a large-scale multi-domain wizard-of-Oz dataset for task-oriented dialogue modelling. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*,

pages 5016–5026, Brussels, Belgium, Oct.-Nov. 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1547. URL https://www.aclweb.org/anthology/D18-1547.

F. Burlot and F. Yvon. Evaluating the morphological competence of machine translation systems. In *Proceedings of the Second Conference on Machine Translation*, pages 43–55, Copenhagen, Denmark, Sept. 2017. Association for Computational Linguistics. doi: 10.18653/v1/W17-4705. URL https://www.aclweb.org/anthology/W17-4705.

T. Caselli, O. Mutlu, A. Basile, and A. Hürriyetoğlu. Protest-er: Retraining bert for protest event extraction. In *Proceedings of the 4th Workshop on Challenges and Applications of Automated Extraction of Socio-political Events from Text (CASE 2021)*, pages 12–19, 2021.

T. Cassidy, B. McDowell, N. Chambers, and S. Bethard. An annotation framework for dense event ordering. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 501–506, Baltimore, Maryland, June 2014. Association for Computational Linguistics. doi: 10.3115/v1/P14-2082. URL https://www.aclweb.org/anthology/P14-2082.

D. Cer, Y. Yang, S.-y. Kong, N. Hua, N. Limtiaco, R. St. John, N. Constant, M. Guajardo-Cespedes, S. Yuan, C. Tar, B. Strope, and R. Kurzweil. Universal sentence encoder for English. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 169–174, Brussels, Belgium, Nov. 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-2029. URL https://www.aclweb.org/anthology/D18-2029.

I. Chalkidis, M. Fergadiotis, S. Kotitsas, P. Malakasiotis, N. Aletras, and I. Androutsopoulos. An empirical study on large-scale multi-label text classification including few and zero-shot labels. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7503–7515, Online, Nov. 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.607. URL https://www.aclweb.org/anthology/2020.emnlp-main.607.

N. Chambers. Navytime: Event and time ordering from raw text. Technical report, Naval Academy Annapolis MD, 2013.

N. Chambers and D. Jurafsky. Jointly combining implicit constraints improves temporal ordering. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 698–706, Honolulu, Hawaii, Oct. 2008. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/D08-1073.

N. Chambers, T. Cassidy, B. McDowell, and S. Bethard. Dense event ordering with a multi-pass architecture. *Transactions of the Association for Computational Linguistics*, 2:273–284, 2014. doi: 10.1162/tacl_a_00182. URL https://www.aclweb.org/anthology/Q14-1022.

Y. S. Chan and H. T. Ng. Estimating class priors in domain adaptation for word sense disambiguation. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 89–96, Sydney, Australia, July 2006. Association for Computational Linguistics. doi: 10.3115/1220175.1220187. URL https://www.aclweb.org/anthology/P06-1012.

Y. S. Chan and H. T. Ng. Domain adaptation with active learning for word sense disambiguation. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 49–56, Prague, Czech Republic, June 2007. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/P07-1007.

A. X. Chang and C. D. Manning. Sutime: A library for recognizing and normalizing time expressions. In *Lrec*, volume 2012, pages 3735–3740, 2012.

M.-W. Chang, M. Connor, and D. Roth. The necessity of combining adaptation methods. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 767–777, Cambridge, MA, Oct. 2010. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/D10-1075.

W. W. Chapman, W. Bridewell, P. Hanbury, G. F. Cooper, and B. G. Buchanan. Evaluation of negation phrases in narrative clinical reports. In *Proceedings of the AMIA Symposium*, page 105. American Medical Informatics Association, 2001.

E. Charniak. Statistical parsing with a context-free grammar and word statistics. *AAAI/IAAI*, 2005 (598-603):18, 1997.

A. Chaudhary, J. Xie, Z. Sheikh, G. Neubig, and J. Carbonell. A little annotation does a lot of good: A study in bootstrapping low-resource named entity recognizers. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5164–5174, Hong Kong, China, Nov. 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1520. URL https://aclanthology.org/D19-1520.

A. Chaudhary, A. Anastasopoulos, Z. Sheikh, and G. Neubig. Reducing confusion in active learning for part-of-speech tagging. *Transactions of the Association for Computational Linguistics*, 9: 1–16, 2021. doi: 10.1162/tacl_a_00350. URL https://aclanthology.org/2021.tacl-1.1.

M. Chen, Z. E. Xu, K. Q. Weinberger, and F. Sha. Marginalized denoising autoencoders for domain adaptation. In *ICML*, 2012.

M. Chen, K. Q. Weinberger, F. Sha, and Y. Bengio. Marginalized denoising auto-encoders for nonlinear representations. In *Proceedings of the 31th International Conference on Machine Learning, ICML 2014, Beijing, China, 21-26 June 2014*, volume 32 of *JMLR Workshop and Conference Proceedings*, pages 1476–1484. JMLR.org, 2014. URL http://proceedings.mlr.press/v32/cheng14.html.

Q. Chen, X. Zhu, Z.-H. Ling, S. Wei, H. Jiang, and D. Inkpen. Enhanced LSTM for natural language inference. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1657–1668, Vancouver, Canada, July 2017a. Association for Computational Linguistics. doi: 10.18653/v1/P17-1152. URL https://www.aclweb.org/anthology/P17-1152.

Q. Chen, X. Zhu, Z.-H. Ling, S. Wei, H. Jiang, and D. Inkpen. Recurrent neural network-based sentence encoder with gated attention for natural language inference. In *Proceedings of the 2nd Workshop on Evaluating Vector Space Representations for NLP*, pages 36–40, Copenhagen, Denmark, Sept. 2017b. Association for Computational Linguistics. doi: 10.18653/v1/W17-5307. URL https://www.aclweb.org/anthology/W17-5307.

Q. Chen, A. Allot, and Z. Lu. Keep up with the latest coronavirus research. *Nature*, 579(7798):193, 2020a. ISSN 1476-4687 (Electronic) 0028-0836 (Linking). doi: 10.1038/d41586-020-00694-1. URL https://www.ncbi.nlm.nih.gov/pubmed/32157233.

Q. Chen, A. Allot, and Z. Lu. Litcovid: an open database of covid-19 literature. *Nucleic acids research*, 49(D1):D1534–D1540, 2021.

S. Chen, Y. Hou, Y. Cui, W. Che, T. Liu, and X. Yu. Recall and learn: Fine-tuning deep pretrained language models with less forgetting. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7870–7881, Online, Nov. 2020b. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.634. URL https://www.aclweb.org/anthology/2020.emnlp-main.634.

T. Chen, Z. Jiang, K. Sakaguchi, and B. Van Durme. Uncertain natural language inference. *arXiv preprint arXiv:1909.03042*, 2019.

W. Chen, J. Chen, Y. Su, X. Wang, D. Yu, X. Yan, and W. Y. Wang. XL-NBT: A cross-lingual neural belief tracking framework. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 414–424, Brussels, Belgium, Oct.-Nov. 2018a. Association for Computational Linguistics. doi: 10.18653/v1/D18-1038. URL https://www.aclweb.org/anthology/D18-1038.

X. Chen, Y. Sun, B. Athiwaratkun, C. Cardie, and K. Weinberger. Adversarial deep averaging networks for cross-lingual sentiment classification. *Transactions of the Association for Computational Linguistics*, 6:557–570, 2018b. doi: 10.1162/tacl_a_00039. URL https://www.aclweb.org/anthology/Q18-1039.

Y. Chen, T. A. Lasko, Q. Mei, J. C. Denny, and H. Xu. A study of active learning methods for named entity recognition in clinical text. *Journal of biomedical informatics*, 58:11–18, 2015.

Y. Chen, T. A. Lask, Q. Mei, Q. Chen, S. Moon, J. Wang, K. Nguyen, T. Dawodu, T. Cohen, J. C. Denny, et al. An active learning-enabled annotation system for clinical named entity recognition. *BMC medical informatics and decision making*, 17(2):35–44, 2017c.

Z. Chen and T. Qian. Transfer capsule network for aspect level sentiment classification. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 547–556, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1052. URL https://www.aclweb.org/anthology/P19-1052.

F. Cheng and Y. Miyao. Classifying temporal relations by bidirectional lstm over dependency paths. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 1–6, Vancouver, Canada, July 2017. Association for Computational Linguistics. URL http://aclweb.org/anthology/P17-2001.

B. Chintagunta, N. Katariya, X. Amatriain, and A. Kannan. Medically aware GPT-3 as a data generator for medical dialogue summarization. In *Proceedings of the Second Workshop on Natural Language Processing for Medical Conversations*, pages 66–76, Online, June 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.nlpmc-1.9. URL https://aclanthology.org/2021.nlpmc-1.9.

L. Chiticariu, R. Krishnamurthy, Y. Li, F. Reiss, and S. Vaithyanathan. Domain adaptation of rule-based annotators for named-entity recognition tasks. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 1002–1012, Cambridge, MA, Oct. 2010. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/D10-1098.

K. Church. A pendulum swung too far. *Linguistic Issues in Language Technology*, 6(5):1–27, 2011.

P. Clark. What knowledge is needed to solve the rte5 textual entailment challenge? *arXiv preprint arXiv:1806.03561*, 2018.

P. Clark, I. Cowhey, O. Etzioni, T. Khot, A. Sabharwal, C. Schoenick, and O. Tafjord. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv preprint arXiv:1803.05457*, 2018.

A. Cocos, A. G. Fiks, and A. J. Masino. Deep learning for pharmacovigilance: recurrent neural network architectures for labeling adverse drug reactions in twitter posts. *J. Am. Medical Informatics Assoc.*, 24(4):813–821, 2017. doi: 10.1093/jamia/ocw180. URL https://doi.org/10.1093/jamia/ocw180.

A. Cohan, S. Feldman, I. Beltagy, D. Downey, and D. Weld. SPECTER: Document-level representation learning using citation-informed transformers. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2270–2282, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.207. URL https://aclanthology.org/2020.acl-main.207.

R. Collobert and J. Weston. A unified architecture for natural language processing: deep neural networks with multitask learning. In W. W. Cohen, A. McCallum, and S. T. Roweis, editors, *Machine Learning, Proceedings of the Twenty-Fifth International Conference (ICML 2008), Helsinki, Finland, June 5-9, 2008*, volume 307 of *ACM International Conference Proceeding Series*, pages 160–167. ACM, 2008. doi: 10.1145/1390156.1390177. URL https://doi.org/10.1145/1390156.1390177.

C. Condoravdi, D. Crouch, V. de Paiva, R. Stolle, and D. G. Bobrow. Entailment, intensionality and text understanding. In *Proceedings of the HLT-NAACL 2003 Workshop on Text Meaning*, pages 38–45, 2003. URL https://www.aclweb.org/anthology/W03-0906.

A. Conneau, D. Kiela, H. Schwenk, L. Barrault, and A. Bordes. Supervised learning of universal sentence representations from natural language inference data. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 670–680, Copenhagen, Denmark, Sept. 2017. Association for Computational Linguistics. doi: 10.18653/v1/D17-1070. URL https://www.aclweb.org/anthology/D17-1070.

A. Conneau, R. Rinott, G. Lample, A. Williams, S. Bowman, H. Schwenk, and V. Stoyanov. XNLI: Evaluating cross-lingual sentence representations. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2475–2485, Brussels, Belgium, Oct.-Nov. 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1269. URL https://www.aclweb.org/anthology/D18-1269.

R. Cooper, D. Crouch, J. Van Eijck, C. Fox, J. Van Genabith, J. Jaspars, H. Kamp, D. Milward, M. Pinkal, M. Poesio, et al. Using the framework. Technical report, 1996.

A. Culotta and A. McCallum. Reducing labeling effort for structured prediction tasks. In *AAAI*, volume 5, pages 746–751, 2005.

A. Cybulska and P. Vossen. Using a sledgehammer to crack a nut? lexical diversity and event coreference resolution. In *Proceedings of the Ninth International Conference on Language*

Resources and Evaluation (LREC'14), pages 4545–4552, Reykjavik, Iceland, May 2014. European Language Resources Association (ELRA). URL http://www.lrec-conf.org/proceedings/lrec2014/pdf/840_Paper.pdf.

I. Dagan and S. P. Engelson. Committee-based sampling for training probabilistic classifiers. In *Machine Learning Proceedings 1995*, pages 150–157. Elsevier, 1995.

I. Dagan, O. Glickman, and B. Magnini. The pascal recognising textual entailment challenge. In *Machine learning challenges. evaluating predictive uncertainty, visual object classification, and recognising tectual entailment*, pages 177–190. Springer, 2006.

I. Dagan, B. Dolan, B. Magnini, and D. Roth. Recognizing textual entailment: Rational, evaluation and approaches. *Natural Language Engineering*, 15(4):i–xvii, 2009.

I. Dagan, B. Dolan, B. Magnini, and D. Roth. The fourth pascal recognizing textual entailment challenge. *Journal of Natural Language Engineering*, 2010.

I. Dagan, D. Roth, M. Sammons, and F. M. Zanzotto. Recognizing textual entailment: Models and applications. *Synthesis Lectures on Human Language Technologies*, 6(4):1–220, 2013.

X. Dai, S. Karimi, B. Hachey, and C. Paris. Using similarity measures to select pretraining data for NER. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1460–1470, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1149. URL https://aclanthology.org/N19-1149.

P. Dasigi, N. F. Liu, A. Marasović, N. A. Smith, and M. Gardner. Quoref: A reading comprehension dataset with questions requiring coreferential reasoning. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5925–5932, Hong Kong, China, Nov. 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1606. URL https://www.aclweb.org/anthology/D19-1606.

H. Daumé III. Frustratingly easy domain adaptation. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 256–263, Prague, Czech Republic, June 2007. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/P07-1033.

M.-C. de Marneffe, A. N. Rafferty, and C. D. Manning. Finding contradictions in text. In *Proceedings of ACL-08: HLT*, pages 1039–1047, Columbus, Ohio, June 2008. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/P08-1118.

M.-C. de Marneffe, S. Padó, and C. D. Manning. Multi-word expressions in textual inference: Much ado about nothing? In *Proceedings of the 2009 Workshop on Applied Textual Inference (TextInfer)*, pages 1–9, Suntec, Singapore, Aug. 2009. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/W09-2501.

S. Dehaene. *The number sense: How the mind creates mathematics*. OUP USA, 2011.

P. Denis and P. Muller. Predicting globally-coherent temporal structures from texts via endpoint inference and graph decomposition. In *Twenty-Second International Joint Conference on Artificial Intelligence*, 2011.

N. Dereli and M. Saraclar. Convolutional neural networks for financial text regression. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pages 331–337, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-2046. URL https://www.aclweb.org/anthology/P19-2046.

B. Desmet, J. Porcino, A. Zirikly, D. Newman-Griffis, G. Divita, and E. Rasch. Development of natural language processing tools to support determination of federal disability benefits in the U.S. In *Proceedings of the 1st Workshop on Language Technologies for Government and Public Administration (LT4Gov)*, pages 1–6, Marseille, France, May 2020. European Language Resources Association. ISBN 979-10-95546-62-7. URL https://aclanthology.org/2020.lt4gov-1.1.

J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1423. URL https://www.aclweb.org/anthology/N19-1423.

P. Dhillon, P. Talukdar, and K. Crammer. Metric learning for graph-based domain adaptation. In *Proceedings of COLING 2012: Posters*, pages 255–264, Mumbai, India, Dec. 2012. The COLING 2012 Organizing Committee. URL https://www.aclweb.org/anthology/C12-2026.

Q. Do, W. Lu, and D. Roth. Joint inference for event timeline construction. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 677–687, Jeju Island, Korea, July 2012. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/D12-1062.

G. Doddington, A. Mitchell, M. Przybocki, L. Ramshaw, S. Strassel, and R. Weischedel. The automatic content extraction (ACE) program – tasks, data, and evaluation. In *Proceedings*

of the Fourth International Conference on Language Resources and Evaluation (LREC'04), Lisbon, Portugal, May 2004. European Language Resources Association (ELRA). URL http://www.lrec-conf.org/proceedings/lrec2004/pdf/5.pdf.

J. Dodge, G. Ilharco, R. Schwartz, A. Farhadi, H. Hajishirzi, and N. A. Smith. Fine-tuning pretrained language models: Weight initializations, data orders, and early stopping. *CoRR*, abs/2002.06305, 2020. URL https://arxiv.org/abs/2002.06305.

D. Dua, Y. Wang, P. Dasigi, G. Stanovsky, S. Singh, and M. Gardner. DROP: A reading comprehension benchmark requiring discrete reasoning over paragraphs. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2368–2378, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1246. URL https://www.aclweb.org/anthology/N19-1246.

L. Duong, T. Cohn, S. Bird, and P. Cook. Cross-lingual transfer for unsupervised dependency parsing without parallel data. In *Proceedings of the Nineteenth Conference on Computational Natural Language Learning*, pages 113–122, Beijing, China, July 2015. Association for Computational Linguistics. doi: 10.18653/v1/K15-1012. URL https://www.aclweb.org/anthology/K15-1012.

L. Duong, H. Afshar, D. Estival, G. Pink, P. Cohen, and M. Johnson. Multilingual semantic parsing and code-switching. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, pages 379–389, Vancouver, Canada, Aug. 2017. Association for Computational Linguistics. doi: 10.18653/v1/K17-1038. URL https://www.aclweb.org/anthology/K17-1038.

J. Ebrahimi, A. Rao, D. Lowd, and D. Dou. HotFlip: White-box adversarial examples for text classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 31–36, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-2006. URL https://www.aclweb.org/anthology/P18-2006.

J. Eisenstein. What to do about bad language on the internet. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 359–369, Atlanta, Georgia, June 2013. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/N13-1037.

R. B. Ekstrom, D. Dermen, and H. H. Harman. *Manual for kit of factor-referenced cognitive tests*, volume 102. Educational Testing Service Princeton, NJ, 1976.

A. Ettinger, S. Rao, H. Daumé III, and E. M. Bender. Towards linguistically generalizable NLP systems: A workshop and shared task. In *Proceedings of the First Workshop on Building*

Linguistically Generalizable NLP Systems, pages 1–10, Copenhagen, Denmark, Sept. 2017. Association for Computational Linguistics. doi: 10.18653/v1/W17-5401. URL https://www.aclweb.org/anthology/W17-5401.

M. Fares, S. Oepen, and E. Velldal. Transfer and multi-task learning for noun–noun compound interpretation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1488–1498, Brussels, Belgium, Oct.-Nov. 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1178. URL https://www.aclweb.org/anthology/D18-1178.

M. Faruqui, J. Dodge, S. K. Jauhar, C. Dyer, E. Hovy, and N. A. Smith. Retrofitting word vectors to semantic lexicons. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1606–1615, Denver, Colorado, May–June 2015. Association for Computational Linguistics. doi: 10.3115/v1/N15-1184. URL https://www.aclweb.org/anthology/N15-1184.

J. R. Finkel and C. D. Manning. Hierarchical Bayesian domain adaptation. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 602–610, Boulder, Colorado, June 2009. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/N09-1068.

N. Fiorini, D. J. Lipman, and Z. Lu. Cutting edge: towards pubmed 2.0. *Elife*, 6:e28801, 2017.

G. Foster, C. Goutte, and R. Kuhn. Discriminative instance weighting for domain adaptation in statistical machine translation. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 451–459, Cambridge, MA, Oct. 2010. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/D10-1044.

M. C. Frank, D. L. Everett, E. Fedorenko, and E. Gibson. Number as a cognitive technology: Evidence from pirahã language and cognition. *Cognition*, 108(3):819–824, 2008.

B. Fu, Z. Cao, J. Wang, and M. Long. Transferable query selection for active domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7272–7281, 2021.

Y. Fyodorov. A natural logic inference system. Citeseer.

Y. Ganin and V. S. Lempitsky. Unsupervised domain adaptation by backpropagation. In F. R. Bach and D. M. Blei, editors, *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, volume 37 of *JMLR Workshop and Conference Proceedings*, pages 1180–1189. JMLR.org, 2015. URL http://proceedings.mlr.press/v37/ganin15.html.

Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. S. Lempitsky. Domain-adversarial training of neural networks. *J. Mach. Learn. Res.*, 17: 59:1–59:35, 2016. URL http://jmlr.org/papers/v17/15-239.html.

J. Gao, J. Lanchantin, M. L. Soffa, and Y. Qi. Black-box generation of adversarial text sequences to evade deep learning classifiers. In *2018 IEEE Security and Privacy Workshops (SPW)*, pages 50–56. IEEE, 2018.

M. Gardner, Y. Artzi, V. Basmova, J. Berant, B. Bogin, S. Chen, P. Dasigi, D. Dua, Y. Elazar, A. Gottumukkala, et al. Evaluating nlp models via contrast sets. *arXiv preprint arXiv:2004.02709*, 2020.

G. Gee and E. Wang. psyML at SemEval-2018 task 1: Transfer learning for sentiment and emotion analysis. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 369–376, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/S18-1056. URL https://www.aclweb.org/anthology/S18-1056.

M. Geva, Y. Goldberg, and J. Berant. Are we modeling the task or the annotator? an investigation of annotator bias in natural language understanding datasets. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1161–1166, Hong Kong, China, Nov. 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1107. URL https://www.aclweb.org/anthology/D19-1107.

R. Ghaeini, S. A. Hasan, V. Datla, J. Liu, K. Lee, A. Qadir, Y. Ling, A. Prakash, X. Fern, and O. Farri. DR-BiLSTM: Dependent reading bidirectional LSTM for natural language inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1460–1469, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-1132. URL https://www.aclweb.org/anthology/N18-1132.

D. Giampiccolo, B. Magnini, I. Dagan, and B. Dolan. The third PASCAL recognizing textual entailment challenge. In *Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*, pages 1–9, Prague, June 2007. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/W07-1401.

R. M. Giménez-Pérez, M. Franco-Salvador, and P. Rosso. Single and cross-domain polarity classification using string kernels. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 558–563, Valencia, Spain, Apr. 2017. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/E17-2089.

O. Glickman, I. Dagan, and M. Koppel. Web based probabilistic textual entailment. 2005.

M. Glockner, V. Shwartz, and Y. Goldberg. Breaking NLI systems with sentences that require simple lexical inferences. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 650–655, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-2103. URL https://www.aclweb.org/anthology/P18-2103.

X. Glorot, A. Bordes, and Y. Bengio. Domain adaptation for large-scale sentiment classification: A deep learning approach. In L. Getoor and T. Scheffer, editors, *Proceedings of the 28th International Conference on Machine Learning, ICML 2011, Bellevue, Washington, USA, June 28 - July 2, 2011*, pages 513–520. Omnipress, 2011. URL https://icml.cc/2011/papers/342_icmlpaper.pdf.

L. Gong, M. Al Boni, and H. Wang. Modeling social norms evolution for personalized sentiment classification. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 855–865, Berlin, Germany, Aug. 2016. Association for Computational Linguistics. doi: 10.18653/v1/P16-1081. URL https://www.aclweb.org/anthology/P16-1081.

I. J. Goodfellow, J. Shlens, and C. Szegedy. Explaining and harnessing adversarial examples. In Y. Bengio and Y. LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. URL http://arxiv.org/abs/1412.6572.

A. Gretton, K. Borgwardt, M. Rasch, B. Schölkopf, and A. Smola. A kernel method for the two-sample-problem. *Advances in neural information processing systems*, 19:513–520, 2006.

H. P. Grice. Logic and conversation. *1975*, pages 41–58, 1975.

R. Grishman and B. Sundheim. Message understanding conference- 6: A brief history. In *COLING 1996 Volume 1: The 16th International Conference on Computational Linguistics*, 1996. URL https://www.aclweb.org/anthology/C96-1079.

Y. Gu, R. Tinn, H. Cheng, M. Lucas, N. Usuyama, X. Liu, T. Naumann, J. Gao, and H. Poon. Domain-specific language model pretraining for biomedical natural language processing. *arXiv preprint arXiv:2007.15779*, 2020.

T. Gui, Q. Zhang, H. Huang, M. Peng, and X. Huang. Part-of-speech tagging for twitter with adversarial neural networks. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2411–2420, Copenhagen, Denmark, Sept. 2017. Association for Computational Linguistics. doi: 10.18653/v1/D17-1256. URL https://www.aclweb.org/anthology/D17-1256.

S. Gururangan, S. Swayamdipta, O. Levy, R. Schwartz, S. Bowman, and N. A. Smith. Annotation artifacts in natural language inference data. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 107–112, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-2017. URL https://www.aclweb.org/anthology/N18-2017.

S. Gururangan, A. Marasović, S. Swayamdipta, K. Lo, I. Beltagy, D. Downey, and N. A. Smith. Don't stop pretraining: Adapt language models to domains and tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8342–8360, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.740. URL https://aclanthology.org/2020.acl-main.740.

S. Gururangan, M. Lewis, A. Holtzman, N. A. Smith, and L. Zettlemoyer. Demix layers: Disentangling domains for modular language modeling. *arXiv preprint arXiv:2108.05036*, 2021.

K. Guu, K. Lee, Z. Tung, P. Pasupat, and M. Chang. Retrieval augmented language model pre-training. In *International Conference on Machine Learning*, pages 3929–3938. PMLR, 2020.

P. Haeusser, A. Mordvintsev, and D. Cremers. Learning by association–a versatile semi-supervised training method for neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 89–98, 2017.

A. Haghighi, A. Ng, and C. Manning. Robust textual inference via graph matching. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pages 387–394, Vancouver, British Columbia, Canada, Oct. 2005. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/H05-1049.

R. Han, Q. Ning, and N. Peng. Joint event and temporal relation extraction with shared representations and structured prediction. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 434–444, Hong Kong, China, Nov. 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1041. URL https://www.aclweb.org/anthology/D19-1041.

X. Han and J. Eisenstein. Unsupervised domain adaptation of contextualized embeddings for sequence labeling. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4238–4248, Hong Kong, China, Nov. 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1433. URL https://www.aclweb.org/anthology/D19-1433.

V. Hangya, F. Braune, A. Fraser, and H. Schütze. Two methods for domain adaptation of bilingual tasks: Delightfully simple and broadly applicable. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 810–820, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-1075. URL https://www.aclweb.org/anthology/P18-1075.

S. Harabagiu and A. Hickl. Methods for using textual entailment in open-domain question answering. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 905–912, Sydney, Australia, July 2006. Association for Computational Linguistics. doi: 10.3115/1220175.1220289. URL https://www.aclweb.org/anthology/P06-1114.

P. H. Hartman and D. H. Owens. How to write software specifications. In *Proceedings of the November 14-16, 1967, Fall Joint Computer Conference*, AFIPS '67 (Fall), pages 779–790, New York, NY, USA, 1967. ACM. doi: 10.1145/1465611.1465713. URL http://doi.acm.org/10.1145/1465611.1465713.

M. A. Hedderich, L. Lange, H. Adel, J. Strötgen, and D. Klakow. A survey on recent approaches for natural language processing in low-resource scenarios. *arXiv preprint arXiv:2010.12309*, 2020.

M. Heilman and N. Madnani. ETS: Domain adaptation and stacking for short answer scoring. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 275–279, Atlanta, Georgia, USA, June 2013. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/S13-2046.

S. Henry, K. Buchan, M. Filannino, A. Stubbs, and Ö. Uzuner. 2018 n2c2 shared task on adverse drug events and medication extraction in electronic health records. *J. Am. Medical Informatics Assoc.*, 27(1):3–12, 2020. doi: 10.1093/jamia/ocz166. URL https://doi.org/10.1093/jamia/ocz166.

K. M. Hermann, T. Kociský, E. Grefenstette, L. Espeholt, W. Kay, M. Suleyman, and P. Blunsom. Teaching machines to read and comprehend. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, pages 1693–1701, 2015. URL http://papers.nips.cc/paper/5945-teaching-machines-to-read-and-comprehend.

T. Hope, A. Amini, D. Wadden, M. van Zuylen, S. Parasa, E. Horvitz, D. Weld, R. Schwartz, and H. Hajishirzi. Extracting a knowledge base of mechanisms from COVID-19 papers. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for*

Computational Linguistics: Human Language Technologies, pages 4489–4503, Online, June 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.naacl-main.355. URL https://aclanthology.org/2021.naacl-main.355.

M. J. Hosseini, H. Hajishirzi, O. Etzioni, and N. Kushman. Learning to solve arithmetic word problems with verb categorization. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 523–533, Doha, Qatar, Oct. 2014. Association for Computational Linguistics. doi: 10.3115/v1/D14-1058. URL https://www.aclweb.org/anthology/D14-1058.

J. Howard and S. Ruder. Universal language model fine-tuning for text classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 328–339, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-1031. URL https://www.aclweb.org/anthology/P18-1031.

J. Hu, S. Ruder, A. Siddhant, G. Neubig, O. Firat, and M. Johnson. Xtreme: A massively multilingual multi-task benchmark for evaluating cross-lingual generalisation. In *International Conference on Machine Learning*, pages 4411–4421. PMLR, 2020.

D. Huang, S. Shi, C.-Y. Lin, and J. Yin. Learning fine-grained expressions to solve math word problems. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 805–814, Copenhagen, Denmark, Sept. 2017. Association for Computational Linguistics. doi: 10.18653/v1/D17-1084. URL https://www.aclweb.org/anthology/D17-1084.

L. Huang, H. Ji, K. Cho, I. Dagan, S. Riedel, and C. Voss. Zero-shot transfer learning for event extraction. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2160–2170, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-1201. URL https://www.aclweb.org/anthology/P18-1201.

Y. J. Huang, J. Lu, S. Kurohashi, and V. Ng. Improving event coreference resolution by learning argument compatibility from unlabeled data. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 785–795, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1085. URL https://www.aclweb.org/anthology/N19-1085.

Y.-Y. Huang and S.-D. Lin. Transferring user interests across websites with unstructured text for cold-start recommendation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 805–814, Austin, Texas, Nov. 2016. Association for

Computational Linguistics. doi: 10.18653/v1/D16-1077. URL https://www.aclweb.org/anthology/D16-1077.

P. Isabelle, C. Cherry, and G. Foster. A challenge set approach to evaluating machine translation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2486–2496, Copenhagen, Denmark, Sept. 2017. Association for Computational Linguistics. doi: 10.18653/v1/D17-1263. URL https://www.aclweb.org/anthology/D17-1263.

M. Iyyer, J. Wieting, K. Gimpel, and L. Zettlemoyer. Adversarial example generation with syntactically controlled paraphrase networks. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1875–1885, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-1170. URL https://www.aclweb.org/anthology/N18-1170.

A. Jain, G. Kasiviswanathan, and R. Huang. Towards accurate event detection in social media: A weakly supervised approach for learning implicit event indicators. In *Proceedings of the 2nd Workshop on Noisy User-generated Text (WNUT)*, pages 70–77, Osaka, Japan, Dec. 2016. The COLING 2016 Organizing Committee. URL https://www.aclweb.org/anthology/W16-3911.

M. Jeong, C.-Y. Lin, and G. G. Lee. Semi-supervised speech act recognition in emails and forums. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 1250–1259, Singapore, Aug. 2009. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/D09-1130.

Y. Ji, G. Zhang, and J. Eisenstein. Closing the gap: Domain adaptation from explicit to implicit discourse relations. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2219–2224, Lisbon, Portugal, Sept. 2015. Association for Computational Linguistics. doi: 10.18653/v1/D15-1264. URL https://www.aclweb.org/anthology/D15-1264.

R. Jia and P. Liang. Adversarial examples for evaluating reading comprehension systems. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2021–2031, Copenhagen, Denmark, Sept. 2017. Association for Computational Linguistics. doi: 10.18653/v1/D17-1215. URL https://www.aclweb.org/anthology/D17-1215.

J. Jiang and C. Zhai. Instance weighting for domain adaptation in NLP. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 264–271, Prague, Czech Republic, June 2007. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/P07-1034.

X. Jiao, F. Wang, and D. Feng. Convolutional neural network for universal sentence embeddings. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2470–2481, Santa Fe, New Mexico, USA, Aug. 2018. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/C18-1209.

V. Jijkoun and M. De Rijke. Recognizing textual entailment: Is word similarity enough? In *Machine Learning Challenges. Evaluating Predictive Uncertainty, Visual Object Classification, and Recognising Tectual Entailment*, pages 449–460. Springer, 2006.

C. Jochim and H. Schütze. Improving citation polarity classification with product reviews. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 42–48, Baltimore, Maryland, June 2014. Association for Computational Linguistics. doi: 10.3115/v1/P14-2008. URL https://www.aclweb.org/anthology/P14-2008.

A. Johnson, P. Karanasou, J. Gaspers, and D. Klakow. Cross-lingual transfer learning for Japanese named entity recognition. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Industry Papers)*, pages 182–189, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-2023. URL https://www.aclweb.org/anthology/N19-2023.

P. Joshi, S. Santy, A. Budhiraja, K. Bali, and M. Choudhury. The state and fate of linguistic diversity and inclusion in the NLP world. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6282–6293, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.560. URL https://www.aclweb.org/anthology/2020.acl-main.560.

A. Joulin, E. Grave, P. Bojanowski, and T. Mikolov. Bag of tricks for efficient text classification. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 427–431, Valencia, Spain, Apr. 2017. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/E17-2068.

A. Kamath, S. Gupta, and V. Carvalho. Reversing gradients in adversarial domain adaptation for question deduplication and textual entailment tasks. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5545–5550, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1556. URL https://www.aclweb.org/anthology/P19-1556.

Y. Karunanayake, U. Thayasivam, and S. Ranathunga. Transfer learning based free-form speech command classification for low-resource languages. In *Proceedings of the 57th Annual Meeting*

of the Association for Computational Linguistics: Student Research Workshop, pages 288–294, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-2040. URL https://aclanthology.org/P19-2040.

A. R. Kashyap, D. Hazarika, M.-Y. Kan, and R. Zimmermann. Domain divergences: a survey and empirical analysis. *arXiv preprint arXiv:2010.12198*, 2020.

D. Kaushik, E. Hovy, and Z. Lipton. Learning the difference that makes a difference with counterfactually-augmented data. In *International Conference on Learning Representations*, 2019.

K. Keith, A. Handler, M. Pinkham, C. Magliozzi, J. McDuffie, and B. O'Connor. Identifying civilians killed by police with distantly supervised entity-event extraction. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1547–1557, Copenhagen, Denmark, Sept. 2017. Association for Computational Linguistics. doi: 10.18653/v1/D17-1163. URL https://www.aclweb.org/anthology/D17-1163.

P. Keung, Y. Lu, J. Salazar, and V. Bhardwaj. Don't use English dev: On the zero-shot cross-lingual evaluation of contextual embeddings. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 549–554, Online, Nov. 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.40. URL https://www.aclweb.org/anthology/2020.emnlp-main.40.

S. Khanuja, S. Dandapat, A. Srinivasan, S. Sitaram, and M. Choudhury. GLUECoS: An evaluation benchmark for code-switched NLP. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3575–3585, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.329. URL https://www.aclweb.org/anthology/2020.acl-main.329.

T. Khot, A. Sabharwal, and P. Clark. Scitail: A textual entailment dataset from science question answering. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

J. Kim, T. Ohta, and J. Tsujii. Corpus annotation for mining biomedical events from literature. *BMC Bioinform.*, 9, 2008. doi: 10.1186/1471-2105-9-10. URL https://doi.org/10.1186/1471-2105-9-10.

J.-D. Kim, T. Ohta, S. Pyysalo, Y. Kano, and J. Tsujii. Overview of BioNLP'09 shared task on event extraction. In *Proceedings of the BioNLP 2009 Workshop Companion Volume for Shared Task*, pages 1–9, Boulder, Colorado, June 2009. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/W09-1401.

J.-K. Kim, Y.-B. Kim, R. Sarikaya, and E. Fosler-Lussier. Cross-lingual transfer learning for POS tagging without cross-lingual resources. In *Proceedings of the 2017 Conference on Empirical*

Methods in Natural Language Processing, pages 2832–2838, Copenhagen, Denmark, Sept. 2017. Association for Computational Linguistics. doi: 10.18653/v1/D17-1302. URL https://www.aclweb.org/anthology/D17-1302.

Y.-B. Kim, K. Stratos, and R. Sarikaya. Frustratingly easy neural domain adaptation. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 387–396, Osaka, Japan, Dec. 2016. The COLING 2016 Organizing Committee. URL https://www.aclweb.org/anthology/C16-1038.

O. Kolomiyets, S. Bethard, and M.-F. Moens. Extracting narrative timelines as temporal dependency structures. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 88–97, Jeju Island, Korea, July 2012. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/P12-1010.

R. Koncel-Kedziorski, H. Hajishirzi, A. Sabharwal, O. Etzioni, and S. D. Ang. Parsing algebraic word problems into equations. *Transactions of the Association for Computational Linguistics*, 3:585–597, 2015. doi: 10.1162/tacl_a_00160. URL https://www.aclweb.org/anthology/Q15-1042.

A. S. Kroch. *The semantics of scope in English.* PhD thesis, Massachusetts Institute of Technology, 1974.

N. Kushman, Y. Artzi, L. Zettlemoyer, and R. Barzilay. Learning to automatically solve algebra word problems. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 271–281, Baltimore, Maryland, June 2014. Association for Computational Linguistics. doi: 10.3115/v1/P14-1026. URL https://www.aclweb.org/anthology/P14-1026.

T. K. Landauer and S. T. Dumais. A solution to plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge. *Psychological review*, 104(2): 211, 1997.

A. Lazaridou, A. Kuncoro, E. Gribovskaya, D. Agrawal, A. Liska, T. Terzi, M. Gimenez, C. de Masson d'Autume, T. Kocisky, S. Ruder, et al. Mind the gap: Assessing temporal generalization in neural language models. *Advances in Neural Information Processing Systems*, 34, 2021.

H. Lee, M. Recasens, A. Chang, M. Surdeanu, and D. Jurafsky. Joint entity and event coreference resolution across documents. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 489–500, Jeju Island, Korea, July 2012. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/D12-1045.

Y.-S. Lee, R. Fernandez Astudillo, T. Naseem, R. Gangi Reddy, R. Florian, and S. Roukos. Pushing the limits of AMR parsing with self-learning. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3208–3214, Online, Nov. 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.findings-emnlp.288. URL https://www.aclweb.org/anthology/2020.findings-emnlp.288.

M. Lesk. Automatic sense disambiguation using machine readable dictionaries: How to tell a pine cone from an ice cream cone. In *Proceedings of the 5th Annual International Conference on Systems Documentation*, SIGDOC '86, pages 24–26, New York, NY, USA, 1986. ACM. ISBN 0-89791-224-1. doi: 10.1145/318723.318728. URL http://doi.acm.org/10.1145/318723.318728.

H. J. Levesque. On our best behaviour. *Artificial Intelligence*, 212:27–35, 2014.

H. J. Levesque, E. Davis, and L. Morgenstern. The winograd schema challenge. In G. Brewka, T. Eiter, and S. A. McIlraith, editors, *Principles of Knowledge Representation and Reasoning: Proceedings of the Thirteenth International Conference, KR 2012, Rome, Italy, June 10-14, 2012*. AAAI Press, 2012. URL http://www.aaai.org/ocs/index.php/KR/KR12/paper/view/4492.

S. C. Levinson. Pragmatics. In *International Encyclopedia of Social and Behavioral Sciences: Vol. 17*, pages 11948–11954. Pergamon, 2001.

D. D. Lewis and W. A. Gale. A sequential algorithm for training text classifiers. In *SIGIR'94*, pages 3–12. Springer, 1994.

P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W.-t. Yih, T. Rocktäschel, et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. *arXiv preprint arXiv:2005.11401*, 2020.

F. Li, S. J. Pan, O. Jin, Q. Yang, and X. Zhu. Cross-domain co-extraction of sentiment and topic lexicons. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 410–419, Jeju Island, Korea, July 2012. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/P12-1043.

F. Li, J. Du, Y. He, H.-Y. Song, M. Madkour, G. Rao, Y. Xiang, Y. Luo, H. Chen, S. Liu, and L. Wang. Time event ontology (teo): to support semantic representation and reasoning of complex temporal relations of clinical events. *Journal of the American Medical Informatics Association*, 27(7):1046–1056, 2020.

J. Li, A. Ritter, C. Cardie, and E. Hovy. Major life event extraction from twitter based on congratulations/condolences speech acts. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1997–2007, Doha, Qatar, Oct.

2014. Association for Computational Linguistics. doi: 10.3115/v1/D14-1214. URL https://www.aclweb.org/anthology/D14-1214.

S. Li, Y. Xue, Z. Wang, and G. Zhou. Active learning for cross-domain sentiment classification. In *Twenty-Third International Joint Conference on Artificial Intelligence*, 2013.

Y. Li, T. Baldwin, and T. Cohn. Semi-supervised stochastic multi-domain learning using variational inference. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1923–1934, Florence, Italy, July 2019a. Association for Computational Linguistics. doi: 10.18653/v1/P19-1186. URL https://www.aclweb.org/anthology/P19-1186.

Z. Li, Y. Zhang, Y. Wei, Y. Wu, and Q. Yang. End-to-end adversarial memory network for cross-domain sentiment classification. In C. Sierra, editor, *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI 2017, Melbourne, Australia, August 19-25, 2017*, pages 2237–2243. ijcai.org, 2017. doi: 10.24963/ijcai.2017/311. URL https://doi.org/10.24963/ijcai.2017/311.

Z. Li, X. Peng, M. Zhang, R. Wang, and L. Si. Semi-supervised domain adaptation for dependency parsing. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2386–2395, Florence, Italy, July 2019b. Association for Computational Linguistics. doi: 10.18653/v1/P19-1229. URL https://www.aclweb.org/anthology/P19-1229.

B. Liang, H. Li, M. Su, P. Bian, X. Li, and W. Shi. Deep text classification can be fooled. In J. Lang, editor, *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018, July 13-19, 2018, Stockholm, Sweden*, pages 4208–4215. ijcai.org, 2018. doi: 10.24963/ijcai.2018/585. URL https://doi.org/10.24963/ijcai.2018/585.

J. Liang, D. Hu, and J. Feng. Do we really need to access the source data? source hypothesis transfer for unsupervised domain adaptation. In *International Conference on Machine Learning*, pages 6028–6039. PMLR, 2020.

S. Liao and R. Grishman. Using prediction from sentential scope to build a pseudo co-testing learner for event extraction. In *Proceedings of 5th International Joint Conference on Natural Language Processing*, pages 714–722, Chiang Mai, Thailand, Nov. 2011. Asian Federation of Natural Language Processing. URL https://www.aclweb.org/anthology/I11-1080.

B. Y. Lin and W. Lu. Neural adaptation layers for cross-domain named entity recognition. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2012–2022, Brussels, Belgium, Oct.-Nov. 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1226. URL https://www.aclweb.org/anthology/D18-1226.

W. Ling, D. Yogatama, C. Dyer, and P. Blunsom. Program induction by rationale generation: Learning to solve and explain algebraic word problems. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 158–167, Vancouver, Canada, July 2017. Association for Computational Linguistics. doi: 10.18653/v1/P17-1015. URL https://www.aclweb.org/anthology/P17-1015.

T. Linzen. How can we accelerate progress towards human-like linguistic generalization? *arXiv preprint arXiv:2005.00955*, 2020.

P. Lison, J. Barnes, A. Hubin, and S. Touileb. Named entity recognition without labelled data: A weak supervision approach. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1518–1533, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.139. URL https://www.aclweb.org/anthology/2020.acl-main.139.

M. Liu, Z. Tu, T. Zhang, T. Su, X. Xu, and Z. Wang. Ltp: A new active learning strategy for crf-based named entity recognition. *Neural Processing Letters*, pages 1–22, 2022.

P. Liu, X. Qiu, and X. Huang. Adversarial multi-task learning for text classification. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1–10, Vancouver, Canada, July 2017. Association for Computational Linguistics. doi: 10.18653/v1/P17-1001. URL https://www.aclweb.org/anthology/P17-1001.

P. Liu, W. Yuan, J. Fu, Z. Jiang, H. Hayashi, and G. Neubig. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *arXiv preprint arXiv:2107.13586*, 2021.

X. Liu, J. Gao, X. He, L. Deng, K. Duh, and Y.-y. Wang. Representation learning using multi-task deep neural networks for semantic classification and information retrieval. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 912–921, Denver, Colorado, May–June 2015. Association for Computational Linguistics. doi: 10.3115/v1/N15-1092. URL https://www.aclweb.org/anthology/N15-1092.

X. Liu, P. He, W. Chen, and J. Gao. Multi-task deep neural networks for natural language understanding. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4487–4496, Florence, Italy, July 2019a. Association for Computational Linguistics. doi: 10.18653/v1/P19-1441. URL https://www.aclweb.org/anthology/P19-1441.

Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019b.

H. Llorens, E. Saquete, and B. Navarro. TIPSem (English and Spanish): Evaluating CRFs and semantic roles in TempEval-2. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 284–291, Uppsala, Sweden, July 2010. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/S10-1063.

H. Llorens, N. Chambers, N. UzZaman, N. Mostafazadeh, J. Allen, and J. Pustejovsky. SemEval-2015 task 5: QA TempEval - evaluating temporal information understanding with question answering. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 792–800, Denver, Colorado, June 2015. Association for Computational Linguistics. doi: 10.18653/v1/S15-2134. URL https://www.aclweb.org/anthology/S15-2134.

K. Lo, L. L. Wang, M. Neumann, R. Kinney, and D. Weld. S2ORC: The semantic scholar open research corpus. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4969–4983, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.447. URL https://www.aclweb.org/anthology/2020.acl-main.447.

P. LoBue and A. Yates. Types of common-sense knowledge needed for recognizing textual entailment. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 329–334, Portland, Oregon, USA, June 2011. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/P11-2057.

D. Lowell, Z. C. Lipton, and B. C. Wallace. Practical obstacles to deploying active learning. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 21–30, Hong Kong, China, Nov. 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1003. URL https://aclanthology.org/D19-1003.

K. Lu, P. Mardziel, F. Wu, P. Amancharla, and A. Datta. Gender bias in neural natural language processing. *arXiv preprint arXiv:1807.11714*, 2018.

K. Lybarger, M. Ostendorf, and M. Yetisgen. Annotating social determinants of health using active learning, and characterizing determinants using neural event extraction. *Journal of Biomedical Informatics*, 113:103631, 2021.

B. MacCartney. *Natural language inference*. Stanford University, 2009.

P. Malakasiotis and I. Androutsopoulos. Learning textual entailment using SVMs and string similarity measures. In *Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*, pages 42–47, Prague, June 2007. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/W07-1407.

C. Manning, M. Surdeanu, J. Bauer, J. Finkel, S. Bethard, and D. McClosky. The Stanford CoreNLP natural language processing toolkit. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 55–60, Baltimore, Maryland, June 2014. Association for Computational Linguistics. doi: 10.3115/v1/P14-5010. URL https://www.aclweb.org/anthology/P14-5010.

M. P. Marcus, B. Santorini, and M. A. Marcinkiewicz. Building a large annotated corpus of English: The Penn Treebank. *Computational Linguistics*, 19(2):313–330, 1993. URL https://www.aclweb.org/anthology/J93-2004.

M. Marelli, L. Bentivogli, M. Baroni, R. Bernardi, S. Menini, and R. Zamparelli. SemEval-2014 task 1: Evaluation of compositional distributional semantic models on full sentences through semantic relatedness and textual entailment. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 1–8, Dublin, Ireland, Aug. 2014a. Association for Computational Linguistics. doi: 10.3115/v1/S14-2001. URL https://www.aclweb.org/anthology/S14-2001.

M. Marelli, S. Menini, M. Baroni, L. Bentivogli, R. Bernardi, and R. Zamparelli. A SICK cure for the evaluation of compositional distributional semantic models. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 216–223, Reykjavik, Iceland, May 2014b. European Language Resources Association (ELRA). URL http://www.lrec-conf.org/proceedings/lrec2014/pdf/363_Paper.pdf.

R. Marvin and T. Linzen. Targeted syntactic evaluation of language models. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1192–1202, Brussels, Belgium, Oct.-Nov. 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1151. URL https://www.aclweb.org/anthology/D18-1151.

A. McCallum and K. Nigam. Employing em and pool-based active learning for text classification. In *Proceedings of the Fifteenth International Conference on Machine Learning*, pages 350–358, 1998.

B. McCann, N. S. Keskar, C. Xiong, and R. Socher. The natural language decathlon: Multitask learning as question answering. *arXiv preprint arXiv:1806.08730*, 2018.

J. McCarthy. Actions and other events in situation calculus. In D. Fensel, F. Giunchiglia, D. L. McGuinness, and M. Williams, editors, *Proceedings of the Eights International Conference on Principles and Knowledge Representation and Reasoning (KR-02), Toulouse, France, April 22-25, 2002*, pages 615–628. Morgan Kaufmann, 2002.

D. McClosky, E. Charniak, and M. Johnson. Effective self-training for parsing. In *Proceedings of the Human Language Technology Conference of the NAACL, Main Conference*, pages 152–159,

New York City, USA, June 2006. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/N06-1020.

D. McClosky, E. Charniak, and M. Johnson. Automatic domain adaptation for parsing. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 28–36, Los Angeles, California, June 2010. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/N10-1004.

T. McCoy, E. Pavlick, and T. Linzen. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3428–3448, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1334. URL https://www.aclweb.org/anthology/P19-1334.

P. Melville and R. J. Mooney. Diverse ensembles for active learning. In *Proceedings of the twenty-first international conference on Machine learning*, page 74, 2004.

G. A. Miller. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41, 1995.

P. Mirza and S. Tonelli. On the contribution of word embeddings to temporal relation classification. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 2818–2828, Osaka, Japan, Dec. 2016. The COLING 2016 Organizing Committee. URL https://www.aclweb.org/anthology/C16-1265.

T. Mitamura, Y. Yamakawa, S. Holm, Z. Song, A. Bies, S. Kulick, and S. Strassel. Event nugget annotation: Processes and issues. In *Proceedings of the The 3rd Workshop on EVENTS: Definition, Detection, Coreference, and Representation*, pages 66–76, Denver, Colorado, June 2015. Association for Computational Linguistics. doi: 10.3115/v1/W15-0809. URL https://www.aclweb.org/anthology/W15-0809.

T. Mitamura, Z. Liu, and E. H. Hovy. Overview of TAC-KBP 2016 event nugget track. In *Proceedings of the 2016 Text Analysis Conference, TAC 2016, Gaithersburg, Maryland, USA, November 14-15, 2016*. NIST, 2016. URL https://tac.nist.gov/publications/2016/additional.papers/TAC2016.KBP_Event_Nugget_overview.proceedings.pdf.

A. Mitra and C. Baral. Learning to use formulas to solve simple arithmetic problems. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2144–2153, Berlin, Germany, Aug. 2016. Association for Computational Linguistics. doi: 10.18653/v1/P16-1202. URL https://www.aclweb.org/anthology/P16-1202.

B. Mohit, N. Schneider, R. Bhowmick, K. Oflazer, and N. A. Smith. Recall-oriented learning of named entities in Arabic Wikipedia. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 162–173, Avignon, France, Apr. 2012. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/E12-1017.

W. Monroe, S. Green, and C. D. Manning. Word segmentation of informal Arabic with domain adaptation. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 206–211, Baltimore, Maryland, June 2014. Association for Computational Linguistics. doi: 10.3115/v1/P14-2034. URL https://www.aclweb.org/anthology/P14-2034.

V. Moscati. *The scope of negation*. PhD thesis, Università degli Studi di Siena, 2006.

L. Mou, Z. Meng, R. Yan, G. Li, Y. Xu, L. Zhang, and Z. Jin. How transferable are neural networks in NLP applications? In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 479–489, Austin, Texas, Nov. 2016. Association for Computational Linguistics. doi: 10.18653/v1/D16-1046. URL https://www.aclweb.org/anthology/D16-1046.

V. L. Muehleisen. *Antonymy and semantic range in english*. na, 1997.

M. L. Murphy. *Semantic relations and the lexicon: Antonymy, synonymy and other paradigms*. Cambridge University Press, 2003.

A. Naik and C. Rose. Towards open domain event trigger identification using adversarial domain adaptation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7618–7624, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.681. URL https://www.aclweb.org/anthology/2020.acl-main.681.

A. Naik and C. Rosé. Towards open domain event trigger identification using adversarial domain adaptation. *arXiv preprint arXiv:2005.11355*, 2020.

A. Naik, C. Bogart, and C. Rose. Extracting personal medical events for user timeline construction using minimal supervision. In *BioNLP 2017*, pages 356–364, Vancouver, Canada,, Aug. 2017. Association for Computational Linguistics. doi: 10.18653/v1/W17-2346. URL https://www.aclweb.org/anthology/W17-2346.

A. Naik, A. Ravichander, N. Sadeh, C. Rose, and G. Neubig. Stress test evaluation for natural language inference. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2340–2353, Santa Fe, New Mexico, USA, Aug. 2018. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/C18-1198.

A. Naik, L. Breitfeller, and C. Rose. TDDiscourse: A dataset for discourse-level temporal ordering of events. In *Proceedings of the 20th Annual SIGdial Meeting on Discourse and Dialogue*, pages 239–249, Stockholm, Sweden, Sept. 2019. Association for Computational Linguistics. doi: 10.18653/v1/W19-5929. URL https://www.aclweb.org/anthology/W19-5929.

A. Naik, J. Lehman, and C. Rose. Adapting to the long tail: A meta-analysis of transfer learning research for language understanding tasks. *arXiv preprint arXiv:2111.01340*, 2021a.

A. Naik, J. F. Lehman, and C. Rose. Adapting event extractors to medical data: Bridging the covariate shift. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 2963–2975, Online, Apr. 2021b. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/2021.eacl-main.258.

A. Naik, S. Parasa, S. Feldman, L. L. Wang, and T. Hope. Literature-augmented clinical outcome prediction. *arXiv preprint arXiv:2111.08374*, 2021c.

N. Nangia, A. Williams, A. Lazaridou, and S. Bowman. The repeval 2017 shared task: Multi-genre natural language inference with sentence representations. In *Proceedings of the 2nd Workshop on Evaluating Vector Space Representations for NLP*, pages 1–10, Copenhagen, Denmark, September 2017a. Association for Computational Linguistics. URL http://www.aclweb.org/anthology/W17-5301.

N. Nangia, A. Williams, A. Lazaridou, and S. Bowman. The RepEval 2017 shared task: Multi-genre natural language inference with sentence representations. In *Proceedings of the 2nd Workshop on Evaluating Vector Space Representations for NLP*, pages 1–10, Copenhagen, Denmark, Sept. 2017b. Association for Computational Linguistics. doi: 10.18653/v1/W17-5301. URL https://www.aclweb.org/anthology/W17-5301.

W. B. Nelson. *Accelerated testing: statistical models, test plans, and data analysis*, volume 344. John Wiley & Sons, 2009.

D. Newman-Griffis, J. F. Lehman, C. Rosé, and H. Hochheiser. Translational nlp: A new paradigm and general principles for natural language processing research. *arXiv preprint arXiv:2104.07874*, 2021.

M. L. Nguyen, I. W. Tsang, K. M. A. Chai, and H. L. Chieu. Robust domain adaptation for relation extraction via clustering consistency. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 807–817, Baltimore, Maryland, June 2014. Association for Computational Linguistics. doi: 10.3115/v1/P14-1076. URL https://www.aclweb.org/anthology/P14-1076.

T. H. Nguyen and R. Grishman. Event detection and domain adaptation with convolutional neural networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 365–371, Beijing, China, July 2015. Association for Computational Linguistics. doi: 10.3115/v1/P15-2060. URL https://www.aclweb.org/anthology/P15-2060.

Y. Nie and M. Bansal. Shortcut-stacked sentence encoders for multi-domain inference. In *Proceedings of the 2nd Workshop on Evaluating Vector Space Representations for NLP*, pages 41–45, Copenhagen, Denmark, Sept. 2017. Association for Computational Linguistics. doi: 10.18653/v1/W17-5308. URL https://www.aclweb.org/anthology/W17-5308.

Y. Nie, A. Williams, E. Dinan, M. Bansal, J. Weston, and D. Kiela. Adversarial nli: A new benchmark for natural language understanding. *arXiv preprint arXiv:1910.14599*, 2019.

A. Nikfarjam, A. Sarker, K. O'Connor, R. E. Ginn, and G. Gonzalez-Hernandez. Pharmacovigilance from social media: mining adverse drug reaction mentions using sequence labeling with word embedding cluster features. *J. Am. Medical Informatics Assoc.*, 22(3):671–681, 2015. doi: 10.1093/jamia/ocu041. URL https://doi.org/10.1093/jamia/ocu041.

Q. Ning, Z. Feng, and D. Roth. A structured learning approach to temporal relation extraction. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1027–1037, Copenhagen, Denmark, Sept. 2017. Association for Computational Linguistics. doi: 10.18653/v1/D17-1108. URL https://www.aclweb.org/anthology/D17-1108.

Q. Ning, Z. Feng, H. Wu, and D. Roth. Joint reasoning for temporal and causal relations. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2278–2288, Melbourne, Australia, July 2018a. Association for Computational Linguistics. doi: 10.18653/v1/P18-1212. URL https://www.aclweb.org/anthology/P18-1212.

Q. Ning, H. Wu, and D. Roth. A multi-axis annotation scheme for event temporal relations. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1318–1328, Melbourne, Australia, July 2018b. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/P18-1122.

T. O'Gorman, K. Wright-Bettner, and M. Palmer. Richer event description: Integrating event coreference with temporal, causal and bridging annotation. In *Proceedings of the 2nd Workshop on Computing News Storylines (CNS 2016)*, pages 47–56, Austin, Texas, Nov. 2016. Association for Computational Linguistics. doi: 10.18653/v1/W16-5706. URL https://www.aclweb.org/anthology/W16-5706.

F. Olsson. A literature survey of active machine learning in the context of natural language processing. 2009.

B. Onyshkevych, M. E. Okurowski, and L. Carlson. Tasks, domains, and languages for information extraction. In *TIPSTER TEXT PROGRAM: PHASE I: Proceedings of a Workshop held at Fredricksburg, Virginia, September 19-23, 1993*, pages 123–133, Fredericksburg, Virginia, USA, Sept. 1993. Association for Computational Linguistics. doi: 10.3115/1119149.1119165. URL https://www.aclweb.org/anthology/X93-1013.

M. J. Page, J. E. McKenzie, P. M. Bossuyt, I. Boutron, T. C. Hoffmann, C. D. Mulrow, L. Shamseer, J. M. Tetzlaff, E. A. Akl, S. E. Brennan, et al. The prisma 2020 statement: an updated guideline for reporting systematic reviews. *Bmj*, 372, 2021.

F. Pan and J. R. Hobbs. Time in owl-s. In *AAAI Spring Symposium on Semantic Web Services.*, pages 29–36. AAA1, Mar. 2004. URL https://www.aaai.org/Papers/Symposia/Spring/2004/SS-04-06/SS04-06-005.pdf.

N. Papernot, P. D. McDaniel, A. Swami, and R. E. Harang. Crafting adversarial input sequences for recurrent neural networks. In J. Brand, M. C. Valenti, A. Akinpelu, B. T. Doshi, and B. L. Gorsic, editors, *2016 IEEE Military Communications Conference, MILCOM 2016, Baltimore, MD, USA, November 1-3, 2016*, pages 49–54. IEEE, 2016. doi: 10.1109/MILCOM.2016.7795300. URL https://doi.org/10.1109/MILCOM.2016.7795300.

N. Papernot, P. McDaniel, I. Goodfellow, S. Jha, Z. B. Celik, and A. Swami. Practical black-box attacks against machine learning. In *Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security*, pages 506–519. ACM, 2017.

R. J. Passonneau, N. Ide, S. Su, and J. Stuart. Biber redux: Reconsidering dimensions of variation in American English. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 565–576, Dublin, Ireland, Aug. 2014. Dublin City University and Association for Computational Linguistics. URL https://www.aclweb.org/anthology/C14-1054.

Y. Peng, S. Yan, and Z. Lu. Transfer learning in biomedical natural language processing: An evaluation of BERT and ELMo on ten benchmarking datasets. In *Proceedings of the 18th BioNLP Workshop and Shared Task*, pages 58–65, Florence, Italy, Aug. 2019. Association for Computational Linguistics. doi: 10.18653/v1/W19-5006. URL https://www.aclweb.org/anthology/W19-5006.

J. Pennington, R. Socher, and C. Manning. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*

(EMNLP), pages 1532–1543, Doha, Qatar, Oct. 2014. Association for Computational Linguistics. doi: 10.3115/v1/D14-1162. URL https://www.aclweb.org/anthology/D14-1162.

M. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-1202. URL https://www.aclweb.org/anthology/N18-1202.

I. Pilán, E. Volodina, and T. Zesch. Predicting proficiency levels in learner writings by transferring a linguistic complexity model from expert-written coursebooks. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 2101–2111, Osaka, Japan, Dec. 2016. The COLING 2016 Organizing Committee. URL https://www.aclweb.org/anthology/C16-1198.

B. Plank. What to do about non-standard (or non-canonical) language in NLP. In S. Dipper, F. Neubarth, and H. Zinsmeister, editors, *Proceedings of the 13th Conference on Natural Language Processing, KONVENS 2016, Bochum, Germany, September 19-21, 2016*, volume 16 of *Bochumer Linguistische Arbeitsberichte*, 2016. URL https://www.linguistics.rub.de/konvens16/pub/2_konvensproc.pdf.

B. Plank and A. Moschitti. Embedding semantic similarity in tree kernels for domain adaptation of relation extraction. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1498–1507, Sofia, Bulgaria, Aug. 2013. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/P13-1147.

B. Plank and G. van Noord. Effective measures of domain similarity for parsing. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 1566–1576, Portland, Oregon, USA, June 2011. Association for Computational Linguistics. URL https://aclanthology.org/P11-1157.

B. Plank, A. Johannsen, and A. Søgaard. Importance weighting and unsupervised domain adaptation of POS taggers: a negative result. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 968–973, Doha, Qatar, Oct. 2014. Association for Computational Linguistics. doi: 10.3115/v1/D14-1104. URL https://www.aclweb.org/anthology/D14-1104.

A. Poliak, J. Naradowsky, A. Haldar, R. Rudinger, and B. Van Durme. Hypothesis only baselines in natural language inference. In *Proceedings of the Seventh Joint Conference on Lexical and*

Computational Semantics, pages 180–191, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/S18-2023. URL https://www.aclweb.org/anthology/S18-2023.

R. S. Pressman. *Software engineering: a practitioner's approach*. Palgrave Macmillan, 2005.

P. Prettenhofer and B. Stein. Cross-language text classification using structural correspondence learning. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 1118–1127, Uppsala, Sweden, July 2010. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/P10-1114.

J. Pustejovsky, J. M. Castaño, R. Ingria, R. Saurí, R. J. Gaizauskas, A. Setzer, G. Katz, and D. R. Radev. Timeml: Robust specification of event and temporal expressions in text. In M. T. Maybury, editor, *New Directions in Question Answering, Papers from 2003 AAAI Spring Symposium, Stanford University, Stanford, CA, USA*, pages 28–34. AAAI Press, 2003a.

J. Pustejovsky, P. Hanks, R. Sauri, A. See, R. Gaizauskas, A. Setzer, D. Radev, B. Sundheim, D. Day, L. Ferro, et al. The timebank corpus. In *Corpus linguistics*, volume 2003, page 40. Lancaster, UK., 2003b.

A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever. Improving language understanding by generative pre-training. 2018.

P. Rai, A. Saha, H. Daumé, and S. Venkatasubramanian. Domain adaptation meets active learning. In *Proceedings of the NAACL HLT 2010 Workshop on Active Learning for Natural Language Processing*, pages 27–32, Los Angeles, California, June 2010. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/W10-0104.

P. Rajpurkar, J. Zhang, K. Lopyrev, and P. Liang. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas, Nov. 2016. Association for Computational Linguistics. doi: 10.18653/v1/D16-1264. URL https://www.aclweb.org/anthology/D16-1264.

P. Rajpurkar, R. Jia, and P. Liang. Know what you don't know: Unanswerable questions for SQuAD. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 784–789, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-2124. URL https://www.aclweb.org/anthology/P18-2124.

A. Ramponi and B. Plank. Neural unsupervised domain adaptation in NLP—A survey. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6838–6855, Barcelona, Spain (Online), Dec. 2020. International Committee on Computational Lin-

guistics. doi: 10.18653/v1/2020.coling-main.603. URL https://www.aclweb.org/anthology/2020.coling-main.603.

A. Ravichander, A. Naik, C. Rose, and E. Hovy. EQUATE: A benchmark evaluation framework for quantitative reasoning in natural language inference. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 349–361, Hong Kong, China, Nov. 2019. Association for Computational Linguistics. doi: 10.18653/v1/K19-1033. URL https://www.aclweb.org/anthology/K19-1033.

N. Reimers, N. Dehghani, and I. Gurevych. Temporal anchoring of events for the TimeBank corpus. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2195–2204, Berlin, Germany, Aug. 2016. Association for Computational Linguistics. doi: 10.18653/v1/P16-1207. URL https://www.aclweb.org/anthology/P16-1207.

N. Reimers, N. Dehghani, and I. Gurevych. Event time extraction with a decision tree of neural classifiers. *Transactions of the Association for Computational Linguistics*, 6:77–89, 2018. doi: 10.1162/tacl_a_00006. URL https://www.aclweb.org/anthology/Q18-1006.

M. T. Ribeiro, S. Singh, and C. Guestrin. Semantically equivalent adversarial rules for debugging NLP models. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 856–865, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-1079. URL https://www.aclweb.org/anthology/P18-1079.

M. T. Ribeiro, T. Wu, C. Guestrin, and S. Singh. Beyond accuracy: Behavioral testing of nlp models with checklist. *arXiv preprint arXiv:2005.04118*, 2020.

L. Rimell, S. Clark, and M. Steedman. Unbounded dependency recovery for parser evaluation. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 813–821, Singapore, Aug. 2009. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/D09-1085.

A. Ritter, Mausam, O. Etzioni, and S. Clark. Open domain event extraction from twitter. In Q. Yang, D. Agarwal, and J. Pei, editors, *The 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '12, Beijing, China, August 12-16, 2012*, pages 1104–1112. ACM, 2012. doi: 10.1145/2339530.2339704. URL https://doi.org/10.1145/2339530.2339704.

J. D. Rodriguez, A. Caldwell, and A. Liu. Transfer learning for entity recognition of novel classes. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1974–1985, Santa Fe, New Mexico, USA, Aug. 2018. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/C18-1168.

L. Romano, M. Kouylekov, I. Szpektor, I. Dagan, and A. Lavelli. Investigating a generic paraphrase-based approach for relation extraction. In *11th Conference of the European Chapter of the Association for Computational Linguistics*, Trento, Italy, Apr. 2006. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/E06-1052.

A. Romanov and C. Shivade. Lessons from natural language inference in the clinical domain. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1586–1596, Brussels, Belgium, Oct.-Nov. 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1187. URL https://www.aclweb.org/anthology/D18-1187.

S. Roy. *Reasoning about quantities in natural language*. PhD thesis, University of Illinois at Urbana-Champaign, 2017.

S. Roy and D. Roth. Solving general arithmetic word problems. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1743–1752, Lisbon, Portugal, Sept. 2015. Association for Computational Linguistics. doi: 10.18653/v1/D15-1202. URL https://www.aclweb.org/anthology/D15-1202.

S. Ruder, M. E. Peters, S. Swayamdipta, and T. Wolf. Transfer learning in natural language processing. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Tutorials*, pages 15–18, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-5004. URL https://www.aclweb.org/anthology/N19-5004.

R. Rudinger, J. Naradowsky, B. Leonard, and B. Van Durme. Gender bias in coreference resolution. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 8–14, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-2002. URL https://www.aclweb.org/anthology/N18-2002.

M. Sabou, K. Bontcheva, L. Derczynski, and A. Scharl. Corpus annotation through crowdsourcing: Towards best practice guidelines. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 859–866, Reykjavik, Iceland, May 2014. European Language Resources Association (ELRA). URL http://www.lrec-conf.org/proceedings/lrec2014/pdf/497_Paper.pdf.

K. Sakaguchi, K. Duh, M. Post, and B. V. Durme. Robsut wrod reocginiton via semi-character recurrent neural network. In S. P. Singh and S. Markovitch, editors, *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, February 4-9, 2017, San Francisco, California, USA*, pages 3281–3287. AAAI Press, 2017. URL http://aaai.org/ocs/index.php/AAAI/AAAI17/paper/view/14332.

S. Samanta and S. Mehta. Towards crafting text adversarial samples. *arXiv preprint arXiv:1707.02812*, 2017.

M. Sammons, V. Vydiswaran, and D. Roth. "ask not what textual entailment can do for you...". In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 1199–1208, Uppsala, Sweden, July 2010. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/P10-1122.

U. Sapkota, T. Solorio, M. Montes, and S. Bethard. Domain adaptation for authorship attribution: Improved structural correspondence learning. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2226–2235, Berlin, Germany, Aug. 2016. Association for Computational Linguistics. doi: 10.18653/v1/P16-1210. URL https://www.aclweb.org/anthology/P16-1210.

A. Sarker and G. Gonzalez. Portable automatic text classification for adverse drug reaction detection via multi-corpus training. *J. Biomed. Informatics*, 53:196–207, 2015. doi: 10.1016/j.jbi.2014.11.002. URL https://doi.org/10.1016/j.jbi.2014.11.002.

R. Saurí, R. Knippen, M. Verhagen, and J. Pustejovsky. Evita: A robust event recognizer for QA systems. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pages 700–707, Vancouver, British Columbia, Canada, Oct. 2005. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/H05-1088.

T. Scheffer, C. Decomain, and S. Wrobel. Active hidden markov models for information extraction. In *International Symposium on Intelligent Data Analysis*, pages 309–318. Springer, 2001.

C. Scheible and H. Schütze. Sentiment relevance. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 954–963, Sofia, Bulgaria, Aug. 2013. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/P13-1094.

T. Schick and H. Schütze. Exploiting cloze-questions for few-shot text classification and natural language inference. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 255–269, Online, Apr. 2021. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/2021.eacl-main.20.

C. Schröder and A. Niekler. A survey of active learning for text classification using deep neural networks. *arXiv preprint arXiv:2008.07267*, 2020.

F. Schröder and C. Biemann. Estimating the influence of auxiliary tasks for multi-task learning of sequence tagging tasks. In *Proceedings of the 58th Annual Meeting of the Association for*

Computational Linguistics, pages 2971–2985, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.268. URL https://www.aclweb.org/anthology/2020.acl-main.268.

H. Schütze. Automatic word sense discrimination. *Computational Linguistics*, 24(1):97–123, 1998. URL https://www.aclweb.org/anthology/J98-1004.

R. Sennrich. How grammatical is character-level neural machine translation? assessing MT quality with contrastive translation pairs. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 376–382, Valencia, Spain, Apr. 2017. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/E17-2060.

B. Settles. Active learning literature survey. Technical report, University of Wisconsin-Madison Department of Computer Sciences, 2009.

B. Settles and M. Craven. An analysis of active learning strategies for sequence labeling tasks. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 1070–1079, Honolulu, Hawaii, Oct. 2008. Association for Computational Linguistics. URL https://aclanthology.org/D08-1112.

B. Settles, M. Craven, and L. Friedland. Active learning with real annotation costs. In *Proceedings of the NIPS workshop on cost-sensitive learning*, volume 1. Vancouver, CA:, 2008.

A. Setzer. *Temporal information in newswire articles: an annotation scheme and corpus study.* PhD thesis, University of Sheffield, 2002.

H. S. Seung, M. Opper, and H. Sompolinsky. Query by committee. In *Proceedings of the fifth annual workshop on Computational learning theory*, pages 287–294, 1992.

D. Shah, T. Lei, A. Moschitti, S. Romeo, and P. Nakov. Adversarial domain adaptation for duplicate question detection. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1056–1063, Brussels, Belgium, Oct.-Nov. 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1131. URL https://www.aclweb.org/anthology/D18-1131.

C. E. Shannon. A mathematical theory of communication. *The Bell system technical journal*, 27 (3):379–423, 1948.

A. Shelmanov, V. Liventsev, D. Kireev, N. Khromov, A. Panchenko, I. Fedulova, and D. V. Dylov. Active learning with deep pre-trained models for sequence tagging of clinical and biomedical texts. In *2019 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pages 482–489. IEEE, 2019.

X. Shi, W. Fan, and J. Ren. Actively transfer domain knowledge. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 342–357. Springer, 2008.

H. Shimodaira. Improving predictive inference under covariate shift by weighting the log-likelihood function. *Journal of Statistical Planning and Inference*, 90(2):227–244, 2000.

M. Sims, J. H. Park, and D. Bamman. Literary event detection. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3623–3634, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1353. URL https://www.aclweb.org/anthology/P19-1353.

N. A. Smith. Adversarial evaluation for models of natural language. *CoRR*, abs/1207.0245, 2012. URL http://arxiv.org/abs/1207.0245.

A. Søgaard and Y. Goldberg. Deep multi-task learning with low level tasks supervised at lower layers. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 231–235, Berlin, Germany, Aug. 2016. Association for Computational Linguistics. doi: 10.18653/v1/P16-2038. URL https://www.aclweb.org/anthology/P16-2038.

A. Søgaard and M. Haulrich. Sentence-level instance-weighting for graph-based and transition-based dependency parsing. In *Proceedings of the 12th International Conference on Parsing Technologies*, pages 43–47, Dublin, Ireland, Oct. 2011. Association for Computational Linguistics. URL https://aclanthology.org/W11-2906.

Z. Song, A. Bies, S. Strassel, T. Riese, J. Mott, J. Ellis, J. Wright, S. Kulick, N. Ryant, and X. Ma. From light to rich ERE: Annotation of entities, relations, and events. In *Proceedings of the The 3rd Workshop on EVENTS: Definition, Detection, Coreference, and Representation*, pages 89–98, Denver, Colorado, June 2015. Association for Computational Linguistics. doi: 10.3115/v1/W15-0812. URL https://www.aclweb.org/anthology/W15-0812.

R. E. Stafford. Hereditary and environmental components of quantitative reasoning. *Review of Educational Research*, 42(2):183–201, 1972.

M. Steedman, M. Osborne, A. Sarkar, S. Clark, R. Hwa, J. Hockenmaier, P. Ruhlen, S. Baker, and J. Crim. Bootstrapping statistical parsers from small datasets. In *10th Conference of the European Chapter of the Association for Computational Linguistics*, Budapest, Hungary, Apr. 2003. Association for Computational Linguistics. URL https://aclanthology.org/E03-1008.

P. Stenetorp, S. Pyysalo, G. Topić, T. Ohta, S. Ananiadou, and J. Tsujii. brat: a web-based tool for NLP-assisted text annotation. In *Proceedings of the Demonstrations at the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 102–107,

Avignon, France, Apr. 2012. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/E12-2021.

J. Strötgen and M. Gertz. HeidelTime: High quality rule-based extraction and normalization of temporal expressions. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 321–324, Uppsala, Sweden, July 2010. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/S10-1071.

A. Stubbs and Ö. Uzuner. Annotating longitudinal clinical narratives for de-identification: The 2014 i2b2/uthealth corpus. *Journal of biomedical informatics*, 58:S20–S29, 2015.

W. F. Styler IV, S. Bethard, S. Finan, M. Palmer, S. Pradhan, P. C. de Groen, B. Erickson, T. Miller, C. Lin, G. Savova, and J. Pustejovsky. Temporal annotation in the clinical domain. *Transactions of the Association for Computational Linguistics*, 2:143–154, 2014. doi: 10.1162/tacl_a_00172. URL https://www.aclweb.org/anthology/Q14-1012.

J.-C. Su, Y.-H. Tsai, K. Sohn, B. Liu, S. Maji, and M. Chandraker. Active adversarial domain adaptation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 739–748, 2020.

A. Subramanya, S. Petrov, and F. Pereira. Efficient graph-based semi-supervised learning of structured tagging models. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 167–176, Cambridge, MA, Oct. 2010. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/D10-1017.

W. Sun, A. Rumshisky, and O. Uzuner. Evaluating temporal relations in clinical text: 2012 i2b2 challenge. *Journal of the American Medical Informatics Association*, 20(5):806–813, 2013.

G. Szarvas, V. Vincze, R. Farkas, G. Móra, and I. Gurevych. Cross-genre and cross-domain detection of semantic uncertainty. *Computational Linguistics*, 38(2):335–367, 2012. doi: 10.1162/COLI_a_00098. URL https://www.aclweb.org/anthology/J12-2004.

C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. J. Goodfellow, and R. Fergus. Intriguing properties of neural networks. In Y. Bengio and Y. LeCun, editors, *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014. URL http://arxiv.org/abs/1312.6199.

A. Talmor, J. Herzig, N. Lourie, and J. Berant. CommonsenseQA: A question answering challenge targeting commonsense knowledge. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4149–4158, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1421. URL https://www.aclweb.org/anthology/N19-1421.

A. Tamkin, T. Singh, D. Giovanardi, and N. Goodman. Investigating transferability in pretrained language models. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1393–1401, Online, Nov. 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.findings-emnlp.125. URL https://www.aclweb.org/anthology/2020.findings-emnlp.125.

S. Tan and X. Cheng. Improving SCL model for sentiment-transfer learning. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Companion Volume: Short Papers*, pages 181–184, Boulder, Colorado, June 2009. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/N09-2046.

A. Taylor, M. Marcus, and B. Santorini. The penn treebank: an overview. In *Treebanks*, pages 5–22. Springer, 2003.

E. F. Tjong Kim Sang and F. De Meulder. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, pages 142–147, 2003. URL https://www.aclweb.org/anthology/W03-0419.

J. Tourille, O. Ferret, X. Tannier, and A. Névéol. LIMSI-COT at SemEval-2017 task 12: Neural architecture for temporal information extraction from clinical narratives. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 597–602, Vancouver, Canada, Aug. 2017. Association for Computational Linguistics. doi: 10.18653/v1/S17-2098. URL https://www.aclweb.org/anthology/S17-2098.

J. Tretmans. Testing concurrent systems: A formal approach. In *International Conference on Concurrency Theory*, pages 46–65. Springer, 1999.

Y. Tsuboi, H. Kashima, S. Mori, H. Oda, and Y. Matsumoto. Training conditional random fields using incomplete annotations. In *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*, pages 897–904, Manchester, UK, Aug. 2008. Coling 2008 Organizing Committee. URL https://www.aclweb.org/anthology/C08-1113.

S. Umansky-Pesin, R. Reichart, and A. Rappoport. A multi-domain web-based algorithm for POS tagging of unknown words. In *Coling 2010: Posters*, pages 1274–1282, Beijing, China, Aug. 2010. Coling 2010 Organizing Committee. URL https://www.aclweb.org/anthology/C10-2146.

S. Upadhyay, M.-W. Chang, K.-W. Chang, and W.-t. Yih. Learning from explicit and implicit supervision jointly for algebra word problems. In *Proceedings of the 2016 Conference on*

Empirical Methods in Natural Language Processing, pages 297–306, Austin, Texas, Nov. 2016. Association for Computational Linguistics. doi: 10.18653/v1/D16-1029. URL https://www.aclweb.org/anthology/D16-1029.

Ö. Uzuner, Y. Luo, and P. Szolovits. Evaluating the state-of-the-art in automatic de-identification. *Journal of the American Medical Informatics Association*, 14(5):550–563, 2007.

Ö. Uzuner, B. R. South, S. Shen, and S. L. DuVall. 2010 i2b2/va challenge on concepts, assertions, and relations in clinical text. *Journal of the American Medical Informatics Association*, 18(5): 552–556, 2011.

N. UzZaman and J. Allen. TRIPS and TRIOS system for TempEval-2: Extracting temporal information from text. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 276–283, Uppsala, Sweden, July 2010. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/S10-1062.

N. UzZaman, H. Llorens, L. Derczynski, J. Allen, M. Verhagen, and J. Pustejovsky. SemEval-2013 task 1: TempEval-3: Evaluating time expressions, events, and temporal relations. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 1–9, Atlanta, Georgia, USA, June 2013. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/S13-2001.

V. Van Asch and W. Daelemans. Using domain similarity for performance estimation. In *Proceedings of the 2010 Workshop on Domain Adaptation for Natural Language Processing*, pages 31–36, Uppsala, Sweden, July 2010. Association for Computational Linguistics. URL https://aclanthology.org/W10-2605.

A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

M. Verhagen, I. Mani, R. Sauri, J. Littman, S. B. Jang, A. Rumshisky, J. Phillips, and J. Pustejovsky. Automating temporal annotation with tarsqi. In *Proceedings of the ACL Interactive Poster and Demonstration Sessions.*, pages 81–84, Ann Arbor, Michigan, June 2005. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/P05-3021.pdf.

M. Verhagen, R. Gaizauskas, F. Schilder, M. Hepple, G. Katz, and J. Pustejovsky. SemEval-2007 task 15: TempEval temporal relation identification. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*, pages 75–80, Prague, Czech Republic, June 2007. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/S07-1014.

M. Verhagen, R. Saurí, T. Caselli, and J. Pustejovsky. Semeval-2010 task 13: Tempeval-2. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 57–62, Uppsala, Sweden, July 2010. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/S10-1010.

G.-A. Vlad, M.-A. Tanase, C. Onose, and D.-C. Cercel. Sentence-level propaganda detection in news articles with transfer learning and BERT-BiLSTM-capsule model. In *Proceedings of the Second Workshop on Natural Language Processing for Internet Freedom: Censorship, Disinformation, and Propaganda*, pages 148–154, Hong Kong, China, Nov. 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-5022. URL https://www.aclweb.org/anthology/D19-5022.

T. Vu, T. Wang, T. Munkhdalai, A. Sordoni, A. Trischler, A. Mattarella-Micke, S. Maji, and M. Iyyer. Exploring and predicting transferability across NLP tasks. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7882–7926, Online, Nov. 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.635. URL https://www.aclweb.org/anthology/2020.emnlp-main.635.

E. Wallace, S. Feng, N. Kandpal, M. Gardner, and S. Singh. Universal adversarial triggers for attacking and analyzing NLP. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2153–2162, Hong Kong, China, Nov. 2019a. Association for Computational Linguistics. doi: 10.18653/v1/D19-1221. URL https://www.aclweb.org/anthology/D19-1221.

E. Wallace, P. Rodriguez, S. Feng, I. Yamada, and J. Boyd-Graber. Trick me if you can: Human-in-the-loop generation of adversarial examples for question answering. *Transactions of the Association for Computational Linguistics*, 7:387–401, Mar. 2019b. doi: 10.1162/tacl_a_00279. URL https://www.aclweb.org/anthology/Q19-1029.

A. Wang, J. Hula, P. Xia, R. Pappagari, R. T. McCoy, R. Patel, N. Kim, I. Tenney, Y. Huang, K. Yu, S. Jin, B. Chen, B. Van Durme, E. Grave, E. Pavlick, and S. R. Bowman. Can you tell me how to get past sesame street? sentence-level pretraining beyond language modeling. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4465–4476, Florence, Italy, July 2019a. Association for Computational Linguistics. doi: 10.18653/v1/P19-1439. URL https://www.aclweb.org/anthology/P19-1439.

A. Wang, Y. Pruksachatkun, N. Nangia, A. Singh, J. Michael, F. Hill, O. Levy, and S. Bowman. Superglue: A stickier benchmark for general-purpose language understanding systems. In *Advances in Neural Information Processing Systems*, pages 3266–3280, 2019b.

A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, and S. R. Bowman. Glue: A multi-task benchmark and analysis platform for natural language understanding. In *7th International Conference on Learning Representations, ICLR 2019*, 2019c.

L. L. Wang, K. Lo, Y. Chandrasekhar, R. Reas, J. Yang, D. Burdick, D. Eide, K. Funk, Y. Katsis, R. M. Kinney, et al. Cord-19: The covid-19 open research dataset. In *Proceedings of the 1st Workshop on NLP for COVID-19 at ACL 2020*, 2020.

R. Wang, M. Utiyama, L. Liu, K. Chen, and E. Sumita. Instance weighting for neural machine translation domain adaptation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1482–1488, Copenhagen, Denmark, Sept. 2017. Association for Computational Linguistics. doi: 10.18653/v1/D17-1155. URL https://www.aclweb.org/anthology/D17-1155.

Z. Wang, Y. Qu, L. Chen, J. Shen, W. Zhang, S. Zhang, Y. Gao, G. Gu, K. Chen, and Y. Yu. Label-aware double transfer learning for cross-specialty medical named entity recognition. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1–15, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-1001. URL https://www.aclweb.org/anthology/N18-1001.

Z. J. Wang, D. Choi, S. Xu, and D. Yang. Putting humans in the natural language processing loop: A survey. In *Proceedings of the First Workshop on Bridging Human–Computer Interaction and Natural Language Processing*, pages 47–52, Online, Apr. 2021. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/2021.hcinlp-1.8.

A. Warstadt, A. Parrish, H. Liu, A. Mohananey, W. Peng, S.-F. Wang, and S. R. Bowman. Blimp: The benchmark of linguistic minimal pairs for english. *Transactions of the Association for Computational Linguistics*, 8:377–392, 2020.

T. Wattarujeekrit, P. K. Shah, and N. Collier. Pasbio: predicate-argument structures for event extraction in molecular biology. *BMC Bioinform.*, 5:155, 2004. doi: 10.1186/1471-2105-5-155. URL https://doi.org/10.1186/1471-2105-5-155.

J. Weizenbaum. ELIZA - a computer program for the study of natural language communication between man and machine. *Commun. ACM*, 9(1):36–45, 1966. doi: 10.1145/365153.365168. URL https://doi.org/10.1145/365153.365168.

M. Wen, Z. Zheng, H. Jang, G. Xiang, and C. Penstein Rosé. Extracting events with informal temporal references in personal histories in online communities. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 836–842, Sofia, Bulgaria, Aug. 2013. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/P13-2145.

G. Wiedemann, E. Ruppert, and C. Biemann. UHH-LT at SemEval-2019 task 6: Supervised vs. unsupervised transfer learning for offensive language detection. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 782–787, Minneapolis, Minnesota, USA, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/S19-2137. URL https://www.aclweb.org/anthology/S19-2137.

A. Williams, N. Nangia, and S. Bowman. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-1101. URL https://www.aclweb.org/anthology/N18-1101.

T. Winograd. Understanding natural language. *Cognitive psychology*, 3(1):1–191, 1972.

D. Wright and I. Augenstein. Transformer based multi-source domain adaptation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7963–7974, Online, Nov. 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.639. URL https://www.aclweb.org/anthology/2020.emnlp-main.639.

F. Wu, Y. Huang, and J. Yan. Active sentiment domain adaptation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1701–1711, Vancouver, Canada, July 2017. Association for Computational Linguistics. doi: 10.18653/v1/P17-1156. URL https://www.aclweb.org/anthology/P17-1156.

T. Wu, M. T. Ribeiro, J. Heer, and D. Weld. Errudite: Scalable, reproducible, and testable error analysis. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 747–763, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1073. URL https://aclanthology.org/P19-1073.

M. Xia, A. Anastasopoulos, R. Xu, Y. Yang, and G. Neubig. Predicting performance for natural language processing tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8625–8646, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.764. URL https://aclanthology.org/2020.acl-main.764.

B. Xie, L. Yuan, S. Li, C. H. Liu, X. Cheng, and G. Wang. Active learning for domain adaptation: An energy-based approach. *arXiv preprint arXiv:2112.01406*, 2021.

J. Xing, K. Zhu, and S. Zhang. Adaptive multi-task transfer learning for Chinese word segmentation in medical text. In *Proceedings of the 27th International Conference on Computational Linguis-*

tics, pages 3619–3630, Santa Fe, New Mexico, USA, Aug. 2018. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/C18-1307.

Y. Xu, L. Mou, G. Li, Y. Chen, H. Peng, and Z. Jin. Classifying relations via long short term memory networks along shortest dependency paths. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1785–1794, Lisbon, Portugal, Sept. 2015. Association for Computational Linguistics. doi: 10.18653/v1/D15-1206. URL https://www.aclweb.org/anthology/D15-1206.

M. Yan, H. Zhang, D. Jin, and J. T. Zhou. Multi-source meta transfer for low resource multiple-choice question answering. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7331–7341, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.654. URL https://www.aclweb.org/anthology/2020.acl-main.654.

H. Yang, T. Zhuang, and C. Zong. Domain adaptation for syntactic and semantic dependency parsing using deep belief networks. *Transactions of the Association for Computational Linguistics*, 3:271–282, 2015. doi: 10.1162/tacl_a_00138. URL https://www.aclweb.org/anthology/Q15-1020.

Y. Yang and J. Eisenstein. Unsupervised multi-domain adaptation with feature embeddings. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 672–682, Denver, Colorado, May–June 2015. Association for Computational Linguistics. doi: 10.3115/v1/N15-1069. URL https://www.aclweb.org/anthology/N15-1069.

Y. Yang, D. Zhou, and Y. He. An interpretable neural network with topical information for relevant emotion ranking. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3423–3432, Brussels, Belgium, Oct.-Nov. 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1379. URL https://www.aclweb.org/anthology/D18-1379.

Z. Yang, J. Hu, R. Salakhutdinov, and W. Cohen. Semi-supervised QA with generative domain-adaptive nets. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1040–1050, Vancouver, Canada, July 2017. Association for Computational Linguistics. doi: 10.18653/v1/P17-1096. URL https://www.aclweb.org/anthology/P17-1096.

Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. R. Salakhutdinov, and Q. V. Le. Xlnet: Generalized autoregressive pretraining for language understanding. In *Advances in neural information processing systems*, pages 5753–5763, 2019.

K. Yin, A. Moryossef, J. Hochgesang, Y. Goldberg, and M. Alikhani. Including signed languages in natural language processing. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 7347–7360, Online, Aug. 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.570. URL https://aclanthology.org/2021.acl-long.570.

W. Yin, T. Schnabel, and H. Schütze. Online updating of word representations for part-of-speech tagging. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1329–1334, Lisbon, Portugal, Sept. 2015. Association for Computational Linguistics. doi: 10.18653/v1/D15-1155. URL https://www.aclweb.org/anthology/D15-1155.

J. Yu, M. Qiu, J. Jiang, J. Huang, S. Song, W. Chu, and H. Chen. Modelling domain relationships for transfer learning on retrieval-based question answering systems in e-commerce. In Y. Chang, C. Zhai, Y. Liu, and Y. Maarek, editors, *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining, WSDM 2018, Marina Del Rey, CA, USA, February 5-9, 2018*, pages 682–690. ACM, 2018. doi: 10.1145/3159652.3159685. URL https://doi.org/10.1145/3159652.3159685.

N. Yu and S. Kübler. Filling the gap: Semi-supervised learning for opinion detection across domains. In *Proceedings of the Fifteenth Conference on Computational Natural Language Learning*, pages 200–209, Portland, Oregon, USA, June 2011. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/W11-0323.

F. Zanzotto, A. Moschitti, M. Pennacchiotti, and M. Pazienza. Learning textual entailment from examples. In *Second PASCAL recognizing textual entailment challenge*, page 50. PASCAL, 2006.

G. Zarrella and A. Marsh. MITRE at SemEval-2016 task 6: Transfer learning for stance detection. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 458–463, San Diego, California, June 2016. Association for Computational Linguistics. doi: 10.18653/v1/S16-1074. URL https://www.aclweb.org/anthology/S16-1074.

R. Zellers, Y. Bisk, R. Schwartz, and Y. Choi. SWAG: A large-scale adversarial dataset for grounded commonsense inference. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 93–104, Brussels, Belgium, Oct.-Nov. 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1009. URL https://www.aclweb.org/anthology/D18-1009.

R. Zellers, A. Holtzman, Y. Bisk, A. Farhadi, and Y. Choi. HellaSwag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational*

Linguistics, pages 4791–4800, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1472. URL https://www.aclweb.org/anthology/P19-1472.

W. Zellinger, T. Grubinger, E. Lughofer, T. Natschläger, and S. Saminger-Platz. Central moment discrepancy (cmd) for domain-invariant representation learning. *arXiv preprint arXiv:1702.08811*, 2017.

Y. Zhang, R. Barzilay, and T. Jaakkola. Aspect-augmented adversarial networks for domain adaptation. *Transactions of the Association for Computational Linguistics*, 5:515–528, 2017. doi: 10.1162/tacl_a_00077. URL https://www.aclweb.org/anthology/Q17-1036.

J. Zhao, T. Wang, M. Yatskar, V. Ordonez, and K.-W. Chang. Gender bias in coreference resolution: Evaluation and debiasing methods. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 15–20, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-2003. URL https://www.aclweb.org/anthology/N18-2003.

R. Zhong, C. Snell, D. Klein, and J. Steinhardt. Summarizing differences between text distributions with natural language, 2022.

B. Zhou, Q. Ning, D. Khashabi, and D. Roth. Temporal common sense acquisition with minimal supervision. *arXiv preprint arXiv:2005.04304*, 2020.

L. Zhou, S. Dai, and L. Chen. Learn to solve algebra word problems using quadratic programming. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 817–822, Lisbon, Portugal, Sept. 2015. Association for Computational Linguistics. doi: 10.18653/v1/D15-1096. URL https://www.aclweb.org/anthology/D15-1096.

Y. Ziser and R. Reichart. Neural structural correspondence learning for domain adaptation. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, pages 400–410, Vancouver, Canada, Aug. 2017. Association for Computational Linguistics. doi: 10.18653/v1/K17-1040. URL https://www.aclweb.org/anthology/K17-1040.