

Multidocument Text Classification over Heterogeneous Data Sources

James Route

CMU-LTI-25-003

February 2025

Language Technologies Institute
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

Thesis Committee:

Eric Nyberg, Chair
Carolyn Rose
Fernando Diaz
Carolyn Bellamy (USG)

*Submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy
in Language and Information Technology.*

© 2024 James Route

Keywords: natural language processing, neural networks, supervised learning, large language models, text classification, document classification

Multidocument Text Classification over
Heterogeneous Data Sources

by

James Route

A thesis for

PhD

in

Language Technologies

Thesis Committee:

Eric Nyberg, Chair (LTI)

Carolyn Rose (LTI)

Fernando Diaz (LTI)

Carolyn Bellamy (USG)

Carnegie Mellon University
2025

Abstract

This work introduces a class of decision problem modeled on real-world applications where a human expert selects a course of action while drawing on a set of disparate information sources. Our primary contributions target decision makers who are looking to refine existing decision tasks and enable automatic processing. We construct three datasets to reflect these types of decision problems: one dataset uses federal hiring records containing details on applicants to public trust positions, another documents trademark registration applicants and outcomes, and the third consists of resolutions introduced in the US House or Senate for deliberation. Each dataset comprises documents from multiple distinct sources that contain a mix of structured and unstructured contents, as well as time series data that evolve over the course of the decision process. The problems represented by these datasets are of great practical importance to government and industry, but there has not been a systematic study that examines how to approach such problems. We demonstrate that these tasks can be modeled as text classification. A typical text classification approach involves concatenating all data sources into a single document for model input, but we show that this approach has limited effectiveness on complex datasets and that state-of-the-art models may fail to learn the training objective. We explore an ensemble approach that leverages the unique properties of these datasets,

demonstrating that multisource ensemble models outperform state-of-the-art single source baselines. Furthermore, the ensemble approach can be implemented under real-world resource constraints. Because we expect that decision makers who are considering automation will be concerned with the fairness of any solution, we outline a basic first analysis that comprises a series of tests, the results of which can feed into deeper fairness investigations informed by domain expertise. We also consider the application of large language models (LLMs) within the practical resource constraints of our classification experiments, and find that these models contribute to explainability of the dataset but do not improve classification accuracy. We close with a discussion of how to adapt our findings into a repeatable framework that can be applied to other practical decision scenarios. We also explore areas of future experimentation, with a focus on solutions that may arise from relaxing resource limitations and improving methodologies from greater access to domain expertise.

Acknowledgment

First, a deep thanks to my PhD supervisor, Dr. Eric Nyberg, for guidance and throughout this project. His interest and support enabled me to find a unique, practical application as a focus for my work, and to execute it while navigating the challenges of coordinating with an outside organization. I would like to express great appreciation to my thesis committee members, whose ideas for additional paths of exploration led to interesting experiments and sharpened the final work.

Many thanks to my collaborators in the federal government, who were willing to take a chance on my research ideas in the years before LLMs brought AI models into mainstream use. This thanks extends to experts in the decision making domains that I examined, as their time and energy directly contributed to the results in this work. My sincere gratitude to Dr. James Shanahan, whose inspiring coursework, and later, mentorship during the PhD application process, helped start my journey at LTI.

Lastly, a thank you to my family for their unwavering support over the years as I began this project and took it to completion.

Table of Contents

Abstract	i
Acknowledgment	iii
1 Introduction and Background	1
1.1 Motivation	5
2 Related Work	10
2.1 Text Classification	10
2.2 Time Series Classification	15
2.3 Text Classification Datasets	16
3 Datasets and Modeling Approach	18
3.1 Dataset Specification and Selection	20
3.2 Public Trust Dataset	24
3.3 US Patent and Trademark Organization Trademark Case Files	43
3.4 Congressional Resolutions	54
4 Model Overview	62
4.1 Regularized LSTM Classifier	66
4.2 CNN Model with Self-Attention	68
4.3 Handling Time-Series Data	70
5 Experimental Design and Results	74
5.1 Training and Testing Setup	76

5.2	Baselines	77
5.3	Results by Dataset	78
5.4	Conclusion	104
6	Results Exploration	107
6.1	Ensemble Diversity	108
6.2	Ablation Studies	112
6.3	Examining Control Sequences	118
6.4	Actionable Results	123
7	Examining Fairness in Model Decisions	126
7.1	Ensemble of Tests for Fairness	130
7.2	Fairness Evaluations by Dataset	132
7.3	Conclusion	154
8	LLM Experiments	157
8.1	Model Selection	159
8.2	Prompting Strategy and Model Execution	159
8.3	Experimental Design and Results	166
8.4	Conclusion and Further Investigation	170
9	Conclusion	173
9.1	Summary	173
9.2	Future Work	180
	Bibliography	185
	Appendices	194
A	Example Dataset Contents	195
B	Supplementary Experimental Results	203

C	Model Parameters	208
D	LLM Prompts and Responses	210

1 Introduction and Background

This work is focused on predictive modeling for a specific type of decision process that involves a human expert selecting an outcome (often binary) while taking into account multiple sources of information that may be drawn from different domains. Such tasks are commonly encountered in government and industry, and final decisions can have significant effects on involved parties. Some examples of these decision processes are the adjudication of clearances for sensitive positions in government, the awarding of a trademark or patent to a business venture, the approval to sell into a highly regulated market (prescription drugs, initial public offering of stock), and rulings in high-volume legal cases such as immigration or medical claims. These decision processes can be protracted, leading to well-publicized backlogs¹ in areas with a high case volume. Furthermore, there is often a high degree of inconsistency across decision makers in a given domain², which raises existential questions about the validity and fairness of the decision process.

Consequently, there is great practical significance in developing a system that accurately predicts the outcome of a specific decision process when given the same set of inputs that a human expert would use. A strongly performing system offers the benefit of automation, as predictions generated nearly instantaneously can assist or even replace the manual decision making process. An automated system

¹<https://crsreports.congress.gov/product/pdf/IN/IN12463>

²<https://news.stanford.edu/stories/2015/04/deportation-research-hausman-042015>

may also offer more consistency in decisions than a diffuse group of experts who hold different biases and interpretations of the decision guidelines. The combination of greater efficiency and consistency in decisions better serves the end users of these decision processes (commonly the general public); delays to critical services can upend lives, and inconsistent judgments erode trust and legitimacy in public and private institutions.

Given the real-world practicality of potential solutions, we envision key stakeholder groups who would be interested in adapting the results of this thesis, and use these groups to drive the scope and experimental design of this work. Our primary stakeholder group, which we refer to as end-users, comprises owners of relevant decision processes who want to build a predictive system for their own novel datasets. The group's interests are focused on identifying modeling approaches that will achieve prediction accuracy levels that warrant a practical deployment. These stakeholders may be exploring an initial foray into predictive systems, based on the previous examples of decision processes in non-technical areas of government and law, and have limited resources to devote to computing infrastructure. This necessitates a technical approach that emphasizes parsimony in model and experimental design. We also expect this stakeholder group to hold concerns with respect to fairness of a given solution, and will be interested in repeatable processes to identify if a trained model disadvantages a subgroups in the dataset. We also identify a group of secondary stakeholders - model purveyors - who comprise scientists and engineers developing solutions and conducting research on similar problems. Model purveyors will be interested in implementing and extending the

approaches introduced in this paper. They will want to understand the tradeoffs across different technical approaches, particularly when dataset features motivate the choice of one approach over another. They may desire diagnostic tests for fairness that will lead to process improvements, and they may also be interested in applying newer approaches, such as the use of large language models (LLMs), to improve accuracy or explainability.

Throughout this work, we will reference both stakeholder groups to focus discussions on design choices and experimental findings while providing a real-world grounding. Keeping stakeholder needs in mind, this thesis investigates the following research hypotheses:

1. A multisource ensemble of text classifiers will outperform state-of-the-art single source baseline classifiers, and the ensemble can be feasibly implemented with real-world resource constraints.
2. Datasets with greater variety of sources in terms of domain, content (e.g., structured, semi-structured, unstructured data), and length will have a higher performance differential for the ensemble over single source baselines.
3. There will be a significant bias with respect to sensitive variables we identify for each dataset, and this bias will correlate with the actual and predicted outcomes in each dataset.
4. The use of LLMs to augment datasets with decision rationales and external information will increase classifier accuracy while remaining feasible under resource constraints.

The results in this thesis are very positive for both stakeholder groups. For the first stakeholder group, we show that the ensemble significantly outperforms single source baselines on all but the simplest of our three datasets, and it is readily implemented on a single GPU system. We also outline a basic first analysis of fairness and show through a series of three tests that significant biases with respect to the selected sensitive variables exist. Additionally, the ensemble modeling approach offers greater insight into the correlations between sensitive variables and the outcome. We then provide a repeatable framework with practical steps for how a stakeholder would apply our modeling approach to a new dataset, and which requirements need to be met to achieve a successful implementation.

For the second stakeholder group, our results discussion provides detailed insights into dataset characteristics that are most amenable to the ensemble approach, in partial support of hypothesis 2 above. We provide a follow-on discussion to the fairness results detailing actionable steps that a policymaker might take to introduce process improvements that address fairness issues. Although we find that the application of LLMs does not improve classifier accuracy (contrary to hypothesis 4), we explore experimental designs for LLMs that become feasible with the relaxation of resource constraints and will advance understanding of the solution space for this class of decision problem.

The remainder of this document is organized as follows:

- additional background on the classification task and discussion of practical motivations (current chapter)

- a survey of existing work on similar problems (Chapter 2)
- description of the datasets and associated stakeholders (Chapter 3)
- details on the proposed approach, including dataset creation and a discussion of the ensemble model (Chapter 4)
- an outline of the experimental design, with results and an exploration of performance differences across models (Chapter 5)
- further exploration of the datasets to determine characteristics that benefit from the ensemble classifier approach (Chapter 6)
- an evaluation of fairness in the datasets and top performing models, motivated by the real-world grounding of our datasets (Chapter 7)
- experiments in applying a large language model (LLM) to improve task accuracy and explainability (Chapter 8)
- steps for practical deployment of our methodology on new datasets and discussion of future work (Chapter 9)

1.1 Motivation

Our initial work on this class of decision problem centered on a dataset collected during a background check for applicants to government positions that require a public trust clearance. This dataset, which is one of the three examined in this

thesis, offers a good example of practical significance due to well-documented accounts of human error and inefficiency that can be addressed with an automated solution. The background check and evaluation process is resource-intensive and the penalties for failure can be very high, leading to undesirable trade-offs between completion time and confidence in the final outcome. The decision process comprises multiple steps of data collection, verification, and analysis; newly arriving information may substantially change the outcome; and a final decision may need to be rendered using incomplete or uncertain information. A well-trained classifier system can add significant value by making this process faster, more accurate, and potentially more consistent. Although text classification is a well-understood problem, a review of prior work finds that nearly all approaches focus on classification using static inputs that are limited to a single document or source.

There are a broad range of occupations that require public trust clearances such as public health workers, immigration officials, and officers working in finance and contracting. The federal government has experienced repeated difficulties in evaluating applicants to cleared positions at scale, as reported in national media³⁴. Most of the reported challenges are for positions that require a security clearance, which involve a process similar to public trust. These jobs employ a large number of people, all of whom require an evaluation at hiring along with periodic updates: in 2010, the Washington Post estimated that 854,000 people in the US held a

³https://www.washingtonpost.com/business/economy/company-allegedly-misled-government-about-security-clearance-checks/2013/06/27/dfb7ee04-df5c-11e2-b2d4-ea6d8f477a01_story.html

⁴<https://www.pogo.org/analysis/27-and-counting-another-background-check-dumper-pleads-guilty>

top-secret security clearance ⁵. This number has most likely grown over the past decade, and the number of people with lower level clearances, including public trust, is probably several times larger. At such high volumes, the government has had difficulty keeping pace - in 2018, the Government Accountability Office added the personnel security process to its “High Risk List” of functions needing “broad-based transformation or specific reforms,” primarily because the backlog of security assessments reached 700,000 in 2017 ⁶. The pressure of managing a high application volume has also created additional problems. Since 2008, the Department of Justice prosecuted 27 cases against contractors who processed fraudulent assessments, the most famous being a 2014 lawsuit against the contractor USIS ⁷. The lawsuit alleged that the company filed at least 665,000 fraudulent background checks, composing 40 percent of its contracted work, from 2008 to 2012. USIS allegedly used software to improperly mark its background checks as having passed a full review, when either a partial or no review had been performed ⁸. Notably, one of the subjects of a USIS background investigation was Aaron Alexis, a US Navy contractor who killed 12 people in a workplace shooting at the Washington Navy Yard in 2013. Although the lawsuit did not specifically identify the Alexis background check as fraudulent, subsequent press reports indicated a number of red flags in the Alexis case that were missed at the time ⁹. The incident

⁵<https://www.washingtonpost.com/investigations/top-secret-america/2010/07/19/hidden-world-growing-beyond-control-2/>

⁶<https://www.gao.gov/press-release/gao-adds-government-wide-personnel-security-clearance-process-high-risk-list>

⁷<https://www.pogo.org/analysis/2018/07/27-and-counting-another-background-check-dumper-pleads-guilty/>

⁸<https://www.reuters.com/article/us-usa-usis-idUSBREA0M0BD20140123>

⁹<https://www.theguardian.com/world/2013/sep/23/aaron-alexis-navy-yard-gunman->

underscores the stakes involved in background checks and exemplifies a worst-case scenario arising from an overburdened system.

The public trust evaluation process takes place over an extended period of time, and can require months for completion. In general, the process starts once an applicant submits the required biographical data. The evaluators then obtain a standard set of additional information, in the form of discrete documents, from other sources. As these documents arrive, they are reviewed to determine if more information is needed to resolve questions. The presence of significant negative information can also cause the entire case to be canceled. Once all documents are available, there is a review to determine if a case is ready for a final decision or if further information is needed. These steps repeat until the available information is deemed sufficient, and then a final decision is issued. This process is inefficient because it requires repeated manual review of lengthy documents. If significant issues in a case are not identified early, then extra time will be spent late in the process to acquire more information. We expect that it is possible to accurately predict the outcome early in the process with incomplete information, and that we can further refine the prediction as more information arrives. In doing so, we can provide a valuable signal to human evaluators that might lead to more efficient processing, identifying red flags early, and reducing time spent reviewing and triaging documents. The goal of this thesis is to show that there is support for this to be done automatically; we will defer investigations into the effects on real-world processes for future work.

Not much academic research has been done on classifiers for multiple documents with structured, unstructured, and time series contents. As we show in the next section, most related tasks focused on traditional static text classification problems. In addition, the initial practical (i.e., non-academic) approaches to this task that used static text classification did not yield promising results. This is why we believe that a thesis is needed to investigate this task and propose a better solution.

2 Related Work

We find that our classification task is underrepresented in the existing literature. Although we have identified tasks with some similarity to ours, the proposed solutions at best partially fit the problem we are working on. Text classification research tends to focus on single-document tasks, and we have not observed datasets that mix text and time-series data. We have identified a number of prior works that are relevant to various aspects of our problem, and in this section we summarize commonalities while highlighting important contrasts that show how our task is unique.

2.1 Text Classification

Text classification is a fundamental task in NLP, having been a focus of research for several decades. Maron (1961) introduced the Naive Bayes algorithm for automated classification of documents, and Mosteller and Wallace (1963) similarly applied Bayesian analysis to trace the disputed authorship of 12 Federalist papers. Over time, additional machine learning algorithms began to be applied to text classification, including expectation maximization (Nigam et al., 1998), support vector machines (Joachims, 1998b), logistic regression (Schütze et al., 1995), and eventually neural networks, most noticeably using recurrent architectures like long-short term memory (Hochreiter, 1997) and convolutional networks (Collobert

and Weston, 2008). These studies tended to apply machine learning to tasks that focused on a single document, including news articles, websites, emails, and historical texts. Such datasets are useful for testing the efficacy of algorithms and feature representations to encode information on document content and semantics, and they may simulate real-world tasks like spam detection and recommendation engines. Our interest is in extending this work to decision tasks, taken from practical use cases, that require the decision maker to evaluate and assign a label to a collection of documents that contain varying kinds of data.

Text classification over multiple documents is a relatively uncommon task, and the closest analogue to our multi-document work is Lynn et al. (2020) . This work predicted “Big Five” personality traits for Facebook users, and a varying number of each user’s posts were treated as document inputs to the model. The authors adapted the Hierarchical Attention Network model from Yang et al. (2016), using the successive attention layers to first encode representations over words, then over documents, and finally encoding a single user-level vector to then generate the personality prediction. There are key differences between the dataset in Lynn et al. (2020) and our task, which inform the different structure of our approaches. The documents in the Facebook dataset tend to be short, although large in number (an average of 163 documents per user, or 3,619 total words). The documents have a degree of equivalency because they are drawn from the same domain, which comports with using a single model to process all of a user’s messages. In our task, the various document sources are drawn from different domains, convey different information about the subject, and have a time series aspect. This does not con-

form to the design specifications in the Lynn et al. (2020) model, which treats all of the source documents the same (i.e., encodes them with the same GRU layer), which implies that the documents convey similar information about the subject. As a result, we believe that a much different approach is required for our task.

We surveyed many text classification papers to define state-of-the-art performance on single document text classification. Adhikari et al. (2019) provided a useful comparison between the AWD-LSTM model from Merity et al. (2017) adapted to text classification, and several popular classifiers including the hierarchical attention network (Yang et al., 2016), sequence generation model (SGM) from Yang et al. (2018) and multiple CNN-based models. The AWD-LSTM consistently outperformed the other models across four common datasets, and as a result, we select it as one of the individual models for our ensemble while excluding the others. We are also interested in CNN models, as these avoid the cost of backpropagation through time that recurrent layers impose with long input sequences. CNN models with strong performance include Conneau et al. (2016), which incorporates techniques from popular computer vision models including VGG (Simonyan and Zisserman, 2014) and ResNet (He et al., 2016). We also note Xiao and Cho (2016), which introduces a character-level model that offers a hybrid convolution and recurrent layer that offers strong performance and a low parameter count. In general, we find that existing CNN models do not match the top performing recurrent models when comparing results on like datasets. Because of this, we opt to build a new CNN model using techniques from QANet (Yu et al., 2018), originally used in the question answering domain. This model, which we describe in the

Proposed Approach, has architectural features not typically found in CNN-based text classifiers and offers an alternative approach to the common CNN models.

Heavily pretrained models have achieved very low error rates in text classification and other NLP tasks as well. ULMFiT Howard and Ruder (2018) used the AWD-LSTM, but with significant pretraining as a language model on the Wikipedia corpus. The model is then fine-tuned as a language model on a task-specific corpus, and again fine-tuned as a classifier. ULMFiT recorded better scores across multiple datasets than its comparison models. The pretraining and fine-tuning techniques used for ULMFiT may be quite useful for our task, although pretraining on Wikipedia may be of limited use because the form-based text data in our domain is significantly different. BERT (Devlin et al., 2018) was the first of many pretrained transformer models that achieved state of the art results across many tasks. We opt to use BERT as a strong baseline, with the caveat that it has three significant limitations on our task: BERT’s pretraining corpus is out of domain for our task; many of our documents exceed BERT’s limitations on input size; and BERT’s large size make it infeasible for inclusion in our ensemble, due to prohibitive hardware requirements to run multiple BERT models simultaneously. Although larger transformers such as XLNet (Yang et al., 2019) have superseded BERT’s performance while allowing for much longer input documents, the model sizes are impractical for our environment, so we exclude them from our experiments. Dai et al. (2022) and Premasiri et al. (2023) explored techniques for text classification on long documents using transformers, with the latter’s experimental approach of embedding chunks of long documents and fusing the embeddings

together leading to less accurate results than document truncation.

During the course of this work, large language models (LLMs) have arisen to a central focus in NLP research for a wide range of tasks. This includes text classification, although results have been mixed on benchmark datasets in comparison to the encoder-type transformers that established prior state of the art results. Zhang et al. (2024) demonstrated an LLM-based method that outperformed state of the art transformer methods on multiple datasets. Their RGPT approach uses adaptive boosting with a recurrent ensemble of strong learners; the base models included RoBERTa (Liu et al., 2019), LLaMA 2 (Touvron et al., 2023a), and others. Although the results are convincing, such an ensemble is highly computationally intensive and impractical for the constraints (see Chapter 4) that we establish for this work. Sun et al. (2023) introduces an in-context learning method that uses kNN search coupled with few-shot prompting to achieve state-of-the-art results on four benchmark classification datasets. We again note resource limitations, as the work uses a 175 billion parameter model, which again exceeds the resources we are willing to allocate to a solution. Edwards and Camacho-Collados (2024) surveyed nearly 20 text classification datasets, comparing in-context learning approaches using a variety of LLMs with fine-tuned transformers. Their findings revealed that the fine-tuned models - RoBERTa and T5 (Raffel et al., 2023) - outperformed the in-context approaches. We explore the use of LLMs for rationale generation and feature set augmentation in Chapter 8; the high computational cost for the best-performing LLM methods precludes their use in this work.

2.2 Time Series Classification

Ismail Fawaz et al. (2019) produced a survey documenting performance on time series data for nine types of archetypal deep learning models (MLP, ResNet, CNN, CNN with simple attention, etc., plus some variants). The models were tested on 85 univariate and 13 multivariate time series datasets. The survey found that the ResNet architecture consistently outperformed the other model types on univariate and multivariate time series data. We found that Ismail Fawaz et al. (2019) offered the most comprehensive comparison of deep learning models across datasets, although its direct applicability to our task is somewhat limited because all of the datasets are inherently numeric, whereas our dataset combines text with a time dimension and comprises multiple documents per input. However, Ismail Fawaz et al. (2019) offers a useful point of comparison for benchmarking deep models on pure time series data, and we plan to test our models (i.e., the ensemble components) to determine their applicability to purely numeric time series data. Ismail Fawaz et al. (2019) mainly used simpler models, none of which included recurrent architectures. Results for our CNN model, which incorporates residual layers and self-attention, and the regularized LSTM will be complementary to the findings in Ismail Fawaz et al. (2019) and offer insight into how sequential NLP models apply to traditional time series tasks.

Lipton et al. (2015) explored a time series prediction task in the healthcare domain using an LSTM model. The dataset is a multivariate time series with measurements recorded at irregular intervals, and one goal of the classification task

is to generate accurate labels throughout the time series, not just at the final step. One of the training methods for the LSTM involves replicating the target labels across all time steps, generating a prediction for every element in the sequence and computing combined loss for the sequence. The total loss is expressed as a combination of the loss at the final step and the average loss for all steps. The techniques in Lipton et al. (2015) are highly relevant for our time series data, as we also use data recorded at irregular intervals and need to generate predictions throughout the sequence. Although our time series combines numeric and textual data, the details in this work inform our modeling approach.

2.3 Text Classification Datasets

Given that text classification is a well-understood task that has long been studied in NLP, there are many benchmark datasets in wide use today. Common domains include news articles, where the task is often to assign categories to documents, such as in AG News (Zhang et al., 2016), 20 Newsgroups (Lang, 1995), and Reuters (Lewis et al., 2004). Sentiment analysis is a similar task which often involves movie and product reviews, such as SST-2 (Socher et al., 2013) and IMDB (Maas et al., 2011). Other domains include medical texts (Joachims, 1998a), online discussion and comments such as Jigsaw toxicity¹ and SemEval tasks (Barbieri et al., 2018). There are many similar datasets drawn from the aforementioned domains, as well others, such as legal texts. These classification tasks most commonly focus on

¹<https://www.kaggle.com/competitions/jigsaw-toxic-comment-classification-challenge>

text at the document or sentence level, as tasks that involve classification over multiple documents are quite rare. This is a key motivation to create new datasets or compile them from existing public data.

We turn to US government data as a source of repeatable decision tasks that draw on multiple documents or information sources. Of our two public datasets, one is derived from US Patent and Trademark Organization (USPTO) trademark case data. We have not identified NLP or machine learning studies that use this data source. A similar dataset assembled from the USPTO’s patent information (Suzgun et al., 2022) includes annotations for an outcome prediction task. This dataset bears a number of similarities to the trademark data that we compile, but it is missing time series data that we consider to be a key component of our work.

Our other public dataset focuses on predicting whether a congressional resolution will successfully pass. We use data provided by the US Congress. Nay (2017) presents a similar text classification task to predict whether a bill will become law. Nay (2017) uses a different data source², and the task is structured as static, single document classification with no time series aspects. We choose to work with congressional resolutions because they are greater in number and have a more balanced binary outcome. Purpura and Hillard (2006) also performs text classification on congressional legislation, but their task assigns topics to bills, which does not require data to be as broadly scoped.

²<https://www.govtrack.us/>

3 Datasets and Modeling Approach

We selected three datasets that conform to the problem class description outlined in Chapter 1. All contain multiple data sources, which may include structured or unstructured data, and all unfold over time. These datasets comprise the original documents and outcomes for the public trust clearance process, used for vetting applicants for certain types of federal jobs; the trademark application process, as administered by the US Patent and Trademark Organization (USPTO); and congressional resolutions introduced in the US House or Senate. The public trust data are proprietary, which strictly limits the details we can share, but the remaining datasets are openly published and have no such constraints.

We return to the discussion of the key stakeholder groups introduced in Chapter 1 (end users and model purveyors) and describe the desired outcomes for each dataset through the lens of both groups. We envision similar stakeholder needs for the public trust and trademark datasets. End users for these datasets are the government organizations executing the decision processes, and they will be seeking a highly accurate solution that can bring efficiency, automation, and consistency. For example, in the public trust domain, earlier predictions can inform longer-term workforce and hiring planning, which is normally complicated by the uncertainty of the clearance process. End users are likely to be interested in the fairness of a solution and the desire to prevent an algorithm from discriminating

against sensitive groups. Model purveyors, who may include technical staff or affiliated researchers, will be interested in process improvement. If a model accurately predicts a negative outcome early, purveyors may want to alter the decision process to close the case quickly and conserve resources or identify proactive steps to prevent the negative outcome. Similarly, model purveyors will be interested in findings that link key features to an outcome disadvantage for a given subgroup. Such findings may prompt further investigation into the decision process to address outcome discrepancies.

Stakeholders for the congressional resolution dataset are somewhat different, as this dataset involves a political process instead of decisions made in accordance with a specific rubric. Examples of end users include the staff of an elected representative or national party who are interested in political forecasting to inform legislative strategy. An accurate solution may hold value beyond predicting the outcome for a single piece of legislation, because it may enable end users to adjust the parameters of a resolution (sponsors, language, committee referrals, etc.) to see if the predicted outcome changes. Model purveyors may again be experts who implement or iterate on a given solution. They may also include political scientists or other researchers who are interested in quantitative modeling of the legislative process to understand outcomes or sources of bias or are looking to improve their existing models.

There are also key stakeholder-driven limitations on data and solutions for each of the three domains. We anticipate that stakeholders are likely to face practical resource limitations when operationalizing a solution, as none of the

domains are traditionally associated with heavy investment in machine learning. This drives our choice of models toward favoring parsimony and efficiency over simply choosing the largest frontier model available. We also assume that any solution will use imperfect or incomplete information as input. In the public trust domain, for example, there may be additional data sources that are not available to the model due to sensitivity, or because the information becomes available so late in the process that a prediction would no longer hold value. As another example, the outcomes in the congressional dataset are determined by votes, committee actions, or other activities taken by elected officials. These representatives may consider additional information that is outside the scope of the resolution (e.g., the needs of a powerful constituent group) and not accounted for in the dataset. The constraint of imperfect information, which is present each of the datasets, increases the difficulty of the prediction task and should further stress the models that we use.

We treat each of these tasks as text classification problems that specifically illustrate a multi-document scenario. In the sections that follow, we explain the dataset selection criteria and then describe each dataset and its associated preprocessing steps in detail.

3.1 Dataset Specification and Selection

This work focuses on a multi-document text classification scenario. For a given problem, we have a set of n input document sources $[s_1, s_2, \dots, s_n] \in S$. Each

document source can emit one or more documents throughout the duration of the task, where a document d_i consists of words w_j^i . A document source can optionally emit a timestamp t_i and descriptor u_i simultaneously with each d_i , and we refer to such a source $[(t_1, u_1), (t_2, u_2), \dots, (t_n, u_n)] \in T$ as a *control sequence*. Finally, we have the set of possible outcomes for the task $[y_1, y_2, \dots, y_m] \in Y$.

In a typical text classification task, the above description collapses to a single document ($n = 1$) and no control sequence. We would then apply each w_j as a sequential input to a model to develop a vector representation of the document d_1 and issue a prediction \hat{y} . However, as we discussed in Chapter 2, not all text classification tasks collapse to a single document specification, which motivates the introduction of the above framework.

Data sets that meet this specification should contain:

- Multiple data sources that function as separate documents and contain features for use in the classification task
- Document sources that mix structured, semi-structured, or unstructured data
- At least one control sequence that chronicles actions and events (with dates or timestamps) throughout the decision process

Many real-world tasks that fit these requirements tend to use restricted or proprietary data, so we particularly value datasets that are openly published. We also seek to maximize variance across datasets in terms of size, complexity, features, and domain. This helps to generate stronger evidence that our experimental ensemble

approach generalizes beyond a single task. In cases where traditional (baseline) modeling approaches are superior, the results are useful for defining conditions or detection methods to inform an ideal approach for future datasets that also fit the above requirements.

The public trust dataset conforms to most of the key requirements, with the exception that it is non-public. The decision process requires weighing evidence from multiple documents, most of which comprise form data with varying degrees of structure. It also includes a control sequence that provides a comprehensive summary on all actions and events that occurred throughout the decision process.

The trademark dataset has a similar organizational structure with some important differences. It contains one data source that is explicitly a document (the usage statement). The other data sources are more accurately characterized as groups of related features, some of which have been distilled from original documents, according to USPTO information. This dataset also includes a control sequence with the unique properties of covering multiple years or even decades for a single case, as well as including a much larger number of unique actions than the other datasets. The USPTO publishes this dataset openly.

The congressional resolution dataset is smaller and offers less complexity in its features than the other datasets, which also provides a useful contrast. Most of the data sources are unstructured text, but they are typically short in length and follow a consistent pattern. The control sequence is much shorter than in the other datasets, and the events often occur on the same day. This dataset is also openly published.

In using publicly released datasets, we are most likely making a tradeoff in terms of complexity, as only the public trust dataset comprises the original documents used in the decision process. The benefit of non-proprietary data is that we can provide examples of dataset members with all data elements intact. The two datasets we selected are the result of a best effort in searching for relevant tasks among many thousands of openly available, textual datasets. There are certainly other tasks and data that would yield interesting further studies, such as the approval process for prescription drugs or applications from companies for investment funding or initial public offering. However, these data are not available to the public, would require special agreements to obtain, and almost certainly contain privileged information that we could not disclose in a publication. Consequently, we leave the investigation of additional proprietary datasets to future work.

Table 3.1: Dataset Summary

Dataset	N	Avg. Tokens	Document Sources
Public Trust	19197	493	3 semistructured, 1 control sequence
Trademark	59999	1664	1 structured, 1 semistructured, 1 unstructured, 1 control sequence
Resolutions	18750	337	2 semistructured, 1 unstructured, 1 control sequence

3.2 Public Trust Dataset

The public trust dataset consists of unstructured and semi-structured text derived from five different document sources. We work with a sample of approximately 35,000 applicants over a 12 year period. The public trust process is best represented as a pipelined system (Figure 3.1), in which each applicant passes through a common set of discrete steps. Each step in the pipeline is associated with the release of a document from one of the sources. Applicants generally follow the same progression of steps, but depending on an applicant’s circumstances, some steps may be repeated or occur out of order. In addition, applicants may be eliminated from consideration at any one of the steps; as a result, the system does not have all possible documents for all of the applicants. The system also emits a series of states that forms a time-series and also contains the output variables. The time series is commonly referred to as ”milestones” in this domain, which track an applicant’s progress through the pipeline in detail and contain a record of all significant actions and events.

The public trust pipeline (Figure 3.1) generally conforms to the following process: the prospective candidate submits an application (Source A), which includes limited biographical information. A candidate who receives and accepts a formal job offer then submits a detailed biographical history (Source B). After receipt of Source B, the formal evaluation process begins, during which the candidate can be removed from consideration at any point if significant negative information surfaces. Evaluators receive a credit report (Source C) on the candidate, and one or

more due diligence checks (Source D). Multiple instances of these documents may be received depending on the details of the case. Once all information is received, the case evaluator makes a final determination for granting public trust access. A set of milestones with timestamps (Source E) is updated throughout the entire process, providing a detailed accounting of all significant actions and summarizing all communications with the applicant. Sources A and B arrive in order for every applicant, but documents and updates from Sources C-E arrive asynchronously with no guarantee of ordering. A candidate who is eliminated from consideration prior to the final decision may be missing some of these sources, but will always have a document from Source A. Current stakeholders desire an intelligent system that can predict the final milestone (i.e., the output variable) at any point in the process, and we have designed our proposed model to meet this requirement.

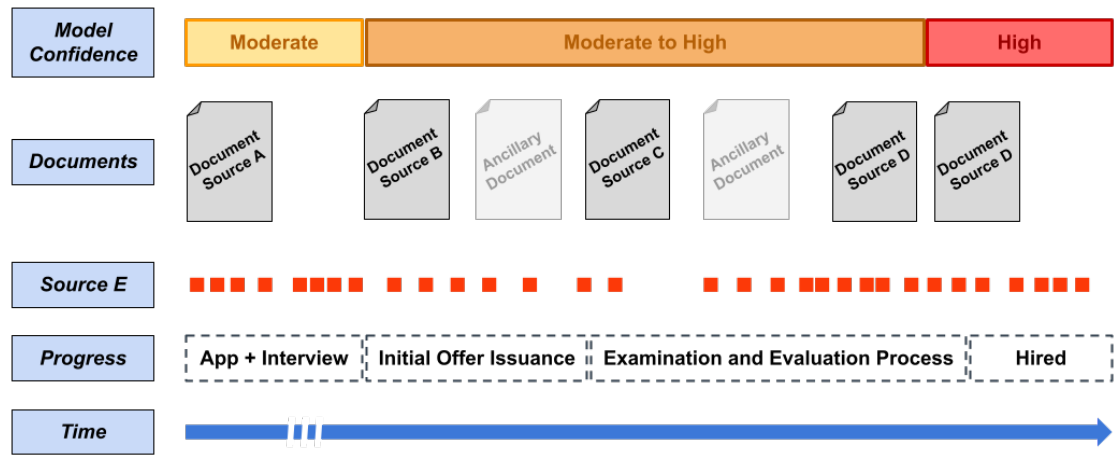


Chart is *not* to scale. Steps may occur out of order. Cancellations or DQs may end process at any point.

Figure 3.1: Summary of the Evaluation Pipeline to Model

We created the dataset by selecting specific data sources and performing ex-

tensive cleaning and standardization, as most sources were designed for human consumption. Our selection process was informed by consultation with subject matter experts who create and review applicant risk evaluations on a daily basis. We also drew on the US Government’s standard rubric that prescribes how to consider and weigh individual risk sources. The raw data sources comprised a combination of relational database tables and document stores. The consultation process was critical to understanding the data sources, narrowing down tables and fields to reasonable sizes, and standardizing sources that evolved multiple times over 12 years. As an example, over 60 relational database tables were relevant to our task, and some tables contained as many as 150 fields.

The following subsections provide a detailed look at each document source. Because of the proprietary nature of the data, we refer to the sources as Document Sources A-E. Although we cannot show specific instances of the public trust data, we provide pseudocode in Appendix A for constructing a notional instance of the public trust dataset. Summary statistics for the data sources are given in Table 3.2. These measures are taken after preprocessing and do not represent raw, unedited documents, which would be much longer. Source C is omitted from the table as it is withheld from use in any model, due to limited coverage in the dataset.

Table 3.2: Field Length Summary for Public Trust Data

Document	Mean	σ	Min	Max
Source A	91.83	23.46	45	259
Source B	184.80	152.22	17	3132
Source D	40.17	16.63	3	140
Source E	176.46	60.00	32	609

3.2.1 Dataset Components

3.2.1.1 Document Source A: Job Application

Document Source A is common to all applicants, and consists of data submitted through an online job application portal. This source is always the first available document in the process, and it is generated only once per applicant. The key document contents include:

- A list of positions the applicant is applying for
- Basic biographical information
- Desired salary
- Employment and residence history
- Self-reported issues, including police record, military disciplinary history, and drug use

- Free-response sections for an objective statement, cover letter, and optional explanatory statements for above issues

We consider this source to be semi-structured data. Although the text is essentially a common form and appears tabular, the employment and residence history sections contain an arbitrary number of entries. The free-response sections are unstructured but vary widely among applicants in length and content.

This source is critically important to prediction because it is always the first document available, and an accurate prediction early in the process carries greater relative value than predictions made later. However, among subject matter experts, this source is considered the most unreliable because applicants are more likely to conceal negative information or provide untruthful answers in Source A than in any of the other documents.

3.2.1.2 Document Source B: Detailed Personal History

Source B is the longest and most complex of the five sources and provides the most comprehensive view of an applicant's background. This source is generated after an applicant accepts a job offer, and its completion marks the start of the formal evaluation process. It is essentially an expansion of Source A, with many additional unique fields added. Key contents include:

- Place of birth, age, citizenship, and similar biographical details
- Comprehensive residence, employment, and educational history

- Negative employment experiences (firings, layoffs, etc.) with optional explanations
- Criminal, drug, alcohol, and mental health histories
- Marital history and information on immediate relatives
- Information on prior government background checks
- Financial history, including debts and negative events such as collections or bankruptcies

Similar to Source A, this document is primarily semi-structured. Although it is an electronic form, many fields contain a variable number of entries, and some optional fields allow for short explanatory responses. One significant departure from Source A is that some applicants submit supplementary free-written pages, mainly providing detailed explanations for negative personal information. The supplementary pages also serve as overflow space when applicants have more entries than the form supports.

The data in Source B is considered more reliable than in Source A. The extra detail that the form requests from applicants, together with a signed statement attesting to the truthfulness of the responses, probably nudges applicants to provide more accurate information.

The documents in Source B can be separated into three groups based on key characteristics. Each group requires varying levels of effort to extract the important fields and standardize the data contents:

- The most recent group, which includes all current and future applicants, requires minimal preprocessing. The data in this group are created from a web-based, interactive form and are stored in JSON format. The web form improves response consistency among applicants and eliminates supplementary material. Approximately 3000 documents are in this format, which is insufficient to train an accurate classifier. In addition, these forms are a biased sample of the applicant population: because the public trust process is time-intensive, the available documents overcount applicants who were quickly disqualified while undercounting applicants who were hired, especially the most complex cases that required extra processing time. We supplement these documents with legacy data to expand our training corpus and alleviate the issue of selection bias.
- The second group comprises 20000 forms originally received in hardcopy, then scanned and sent through OCR software to produce a PDF document. Although the forms contain typed responses, the OCR process introduces numerous errors in the text that need to be resolved during preprocessing.
- The earliest format, approximately 6000 forms, contain handwritten responses, which the OCR process cannot capture.

3.2.1.3 Document Source C: Credit Report

Source C is a standard credit report generated by one of the major credit bureaus. The document is originally received as a PDF (pre-processing details below) and

conforms to one of three formats. The report contents are semi-structured, typically including a text summary that adheres closely to a standard template, a significant amount of boilerplate text, and structured text for each of a subject’s financial accounts. The summary sections across all formats provide an overview of the subject’s personal data (e.g., name, address), totals for accounts opened by type, and records for negative events such as bankruptcies, judgments, collections, and late payments. The details for each account generally include information on the creditor, the amount of credit extended, the balance, alerts for accounts that are in collection or overdue, and payment history (a string encoding the lateness of prior payments).

Depending on an applicant’s financial situation, more than one credit report may be available. We use only the most recent credit report as an input because the earlier reports do not contain unique information. We have credit reports for approximately 30,000 candidates, excluding duplicate or multiple reports for the same subject. As is reported in Chapter 5, the limited coverage for Source C prevents us from using it in experiments; the inclusion of Source C in this document is only to maintain continuity with our previous materials on the public trust task.

3.2.1.4 Document Source D: Due Diligence Reports

Source D is used to corroborate an applicant’s self-reported data, and provides insight into whether an applicant is withholding or distorting information. This data source is terse and follows a structured format; the only deviation from purely

structured data is that each document contains an arbitrary number of entries. When a candidate has multiple documents from Source D, we concatenate the contents using a special token to indicate the start of additional data. The contents of multiple documents are always complementary.

We have documents from Source D for approximately 25,000 candidates. There are two reasons why candidates may be missing Source D information: they were disqualified before any Source D information was available, or they had recently undergone a similar evaluation process with another federal agency.

3.2.1.5 Document Source E: Tracking and Milestones

Source E is unique in that it forms a time-series of short text spans. This source tracks a candidate’s progress through the entire hiring process, beginning with initial application submission and ending with an onboarding date or disqualification. The milestones track every significant event in the process, including receipt of documents from Sources A-D, any correspondence with the candidate, and administrative procedures that are part of the hiring and evaluation process.

Each milestone consists of a timestamp, a title, and an optional comments field. The title is a categorical variable with 30 different levels, and in raw form is a short (< 10 token) span. The comments field usually contains terse, administrative notes which usually have no clear relevance to the evaluation process. In rare circumstances, the comments may contain a summary or verbatim copy of correspondence from a candidate, with an arbitrary text length.

We treat Source E as a single document concatenation of the milestones. Source E is guaranteed to be available for all candidates.

3.2.2 Preprocessing and Dataset Construction

In this section we discuss the process for normalizing each data source prior to the prediction task. Because much of the data are self-reported, there is often significant variance within each data source. Some of the sources (e.g., Source B) have evolved over the 12 year period of our collection, which compounds the challenge of normalization. We focus primarily on enforcing a common schema for all data within each source. For sources that have changed over time, typically the most recent format is more structured and emphasizes a greater number of specific fields over free-text entry. In these cases we treat the most structured format as canonical and perform cleaning and text extraction to transform historical data into a best approximation of the canonical format. We also develop common vocabularies or formats for frequently occurring entity types such as countries, organizations, and dates to ensure that these are expressed in the same way across all documents and sources.

There is also an internal concern about bias in the raw data and the evaluation process. During preprocessing we remove features that could lead to obvious bias or spurious correlations in our model, such as names, gender, age, ethnicity, and detailed location data (e.g., zip codes and street addresses). Many of the other features in the dataset may correlate strongly with one or more protected categories

where bias is often a concern. However, we do not wish to obfuscate sources of bias in features that are of practical use to the evaluation process; we choose to leave these as-is to facilitate further investigation into potential sources of bias.

A final aspect of our preprocessing is the removal of boilerplate text. This typically includes instructions for the subject when filling out a form, administrative details, or other standard text that occurs across most or all of a document set. This text is extraneous to the classification task, provides little to no information, and can easily add thousands of tokens to document length.

The following subsections detail our preprocessing for each dataset and provide rationales for our decisions to reduce data size and feature counts.

3.2.2.1 Preprocessing Document Source A (Job Application)

Although Source A reflects the contents of a prospective employee’s job application, the constituent data is stored in a relational database across 25 different tables. The goal of preprocessing for Source A is to reconstruct relevant parts of the application document and enforce consistency with the other document sources.

We engaged extensively with human experts over the data in Source A to understand which parts of the application were most relevant to evaluations for public trust accesses. We then collected the data from these fields and recreated a flattened, unstructured text document by concatenating the text features and using short boilerplate spans of text to delimit the major sections of the document. Some of the sections in Source A allow for a variable number of responses (e.g.,

employment history may have zero, one, or even tens of entries for an applicant), we delimit each entry with a special token in the document. For the additional features in Source A that were not identified as having the greatest relevance, we experimented with adding these features one at a time and observing if a significant increase in classification accuracy appears relative to baseline over three separate trials with different random seeds. Features with a significant accuracy increase were added to the document, and features with an insignificant or decrease in accuracy were excluded. The most notable features we excluded were free-response sections, including an applicant’s objective statement and optional statements for explaining negative circumstances in one’s background. The inclusion of these fields added to model complexity because they commonly increased vocabulary size by a factor of 2 or 3, but the effect on accuracy was either insignificant or negative. This finding suggested that the unstructured explanatory statements mainly provided noise rather than additional information, which reflected some comments from the human experts. We also remove personal identifying information from Source A, including name, address data (except for country of residence), birth dates and social security numbers. We do not want the classifier to make predictions based on spurious correlations from this data; moreover, some of the information may correlate strongly with race or other protected categories, potentially leading to disparate predictions across these population groups.

We also append metadata for each applicant to Source A, as this information is available at the same point in the decision process and is not large enough to constitute an additional information source. The metadata results from querying

other database tables and consists of a subject’s prior application history and outcomes, whether the subject was a previous employee, and whether the subject had been previously granted or denied public trust accesses.

We describe consistency enforcement in later subsections, as data consistency for Source A is only relevant in comparison to the other data sources.

3.2.2.2 Preprocessing Data Source B (Detailed Personal History)

Source B contains the longest documents in our dataset and presents the largest preprocessing challenge by a wide margin. We partition the Source B documents into three groups, each of which have different preprocessing requirements. There are, however, some preprocessing steps common to all three formats: we sanitize the documents for personally identifying information in the same manner as Source A. We remove sections from Source B that do not directly contribute to the evaluation, such as personal references and certification pages. We currently exclude the optional free-response fields, as these occasionally grow to multi-page documents that mirror responses in other fields. Similar to Source A, we then use the raw semi-structured data to assemble a flattened document that is a simplified version of the original form. We insert text to mark the start of each major section, and use special tokens to delimit consecutive responses in sections that allow for a variable number of response entries. We expect this process to be much less time-intensive for future data, as the new format for Source B does not require a high degree of preprocessing.

Source B, Group 1 Group 1 contains the most recent Source B documents, and it consists of JSON files containing responses entered by an applicant into a web form. The form has internal logic to enforce consistency of responses across applicants and to reduce the incidence of mistakes or typos (e.g., two applicants entering “New York” and “New Yokr” for the same location). We treat this group as the standard for Source B because it contains more specific fields and detailed information than the others. We map preprocessed data from the other groups to match this one. Because the data in Group 1 are clean and consistent, we perform no additional preprocessing.

Source B, Group 2 The second group comprises the majority of the documents in Source B. This group consists of PDFs documents that are at least 15 pages in length, with optional pages available for extra responses. Applicants typed responses into the PDFs and submitted them in hard copy; the printed copies were later scanned and passed through an optical character recognition (OCR) system to recreate the PDF with selectable text. The process of converting the hard copy document into soft copy introduces noise into the text, which complicates our preprocessing.

We developed a tool to extract responses from the PDF and produce a JSON structure that mirrors the format for Group 1. This involves first extracting plain text from the PDF on a page-by-page basis, then identifying the location of key fields and copying responses into the JSON structure. We use pdftools for text extraction (Ooms, 2023), as its output was more consistent across documents than

other open source tools. The printing, scanning, OCR, and PDF-to-plain text conversion processes create an error cascade that greatly complicates this task; as a result, the raw text in the PDF very often does not match what is printed on the page. The complex layout of many of the document pages worsens this problem, and documents that were scanned at an angle contain more inconsistencies. Because we have no alternative source for these documents, our extraction and preprocessing tool needs to be robust to handle the wide range of errors and inconsistencies present. Some examples of common errors are:

- Swapping of adjacent rows of text and adjacent fields within the same row
- Substitution of letters, letter pairs, or numbers with another letter or number that has a similar appearance (e.g., 9 and g).
- Conversion of non-character form elements, such as the icon for a drop-down menu or horizontal and vertical bars used to separate fields, to one or more characters.
- Insertion of random characters, particularly when scan quality is poor or stray marks are present.
- Deletion of characters or spans of text. This is an issue in the OCR process; it occurs frequently in fields with single-character responses, and also when a portion of a document has been crossed out.
- Misordering of pages in the document as a result of handling errors. The document may also contain pages of hand-typed text to supplement the form

contents, as well as various extraneous pages that are of no relevance to our task.

To overcome these issues, we developed a process that scans each page of the document for key text spans to identify which form sections are present. Once the page is identified, we then locate specific text spans that form a boundary around each response that we want to extract. We target text spans that are unique to the targeted page or section and not easily mistaken for text in other parts of the document. We use a multi-step process, falling back to searches with greater tolerance after each failure: we first attempt an exact match, then try a set of regular expressions to match common variants of the target span, and finally identify the closest matching span that is within a maximum Levenshtein distance from the target. We initially selected the target spans, regular expressions, and distance thresholds through an iterative trial-and-error process, with the goal of minimizing errors over a random sample of 10 documents. We then scaled to a random sample 100 documents to further tune our selections. We scaled further to a random sample of 1000 documents, identified the most frequent errors, and adjusted the parameters and target spans to minimize error occurrence.

The extraction process remains imperfect, as some of the errors introduced by the scan/OCR process - particularly the deletions of text spans - can only be rectified through manual review. Our text extractor is designed to fail gracefully, and it records errors in a log file every time it fails to identify a page, locate an extractable span, or it encounters an unexpected condition. Following the automated extraction process, we employ human annotators to quickly review the

extracted result and correct significant errors, such as the omission of fields or entire document sections. We find that annotators, with some practice, can spot-check and correct the extraction results for one document in two to three minutes. With a small team of annotators were able to make steady progress at extracting and cleaning text the full collection of typed PDFs.

Source B, Group 3 The third format comprises the oldest documents in our dataset. Most documents are identical to Group 2 except they contain handwritten responses, which the OCR process cannot capture. There is also a subgroup of 1500-2000 documents in Group 3 that uses an older and slightly more condensed version of the paper form. The response data in both versions are essentially the same, so we do not treat these versions differently. We employ human annotators to transcribe the text in these documents, targeting the same fields as the automated extractor. We developed a custom GUI for annotators to enter transcribed text into, using menus and selectable elements wherever possible to limit typos and other sources of variance. We then write the transcribed data to a JSON, matching the structure of data from Groups 1 and 2. We initially crowd-sourced the annotation task to an internal workforce, and later hired a temporary worker. We were able to complete approximately 2000 of the handwritten forms with these resources; the unfinished forms are from the earliest years in the document collection, which we view as being the least relevant to applicants and decisions for the current pipeline in 2023.

3.2.2.3 Preprocessing Document Source C (Credit Reports)

The credit reports in Source C are electronic PDF files, so no scanning or OCR process is involved. There are three possible formats for the credit reports, each corresponding to one of the three major consumer credit bureaus. The formats convey the same basic information, although their terminology and syntax differ. One format accounts for approximately 50% of the documents, and the other formats have 25% each. There are multiple credit reports available for some applicants, but we use only the most recent report, as it supersedes the previous ones.

The credit reports vary in length, typically up to 10 pages. However, all three formats contain a large amount of boilerplate text at the beginning and end of the document, which we remove. What follows is typically a summary section and then semi-tabular data for each credit account, ordered chronologically or by descending levels of delinquency, depending on the format. Although credit reports contain a significant amount of numeric data, during preliminary testing we found that the contents of the summary section, which is mostly categorical data, and the text annotations that accompany each account and indicate collections, past-due issues, and bankruptcies, are the most relevant to the prediction task. This finding is consistent with the views of expert examiners who review the credit reports. As a result, we simplify numeric values in the credit reports, binning them by order of magnitude and assigning a unique token to each bin.

Similar to Source B, we use the `pdftools` package to extract raw text from the credit reports. This process yields consistent results because the text in these PDFs

is already clean. After removing lengthy boilerplate spans, we remove personally identifying information, including account numbers where possible (some account numbers are nearly indistinguishable from numeric values). We then truncate each document to a maximum length of 900 tokens. We tuned this value by observing effects on classification accuracy, and found that maximum lengths above 900 conferred almost no benefit. Without truncation, documents can exceed 3000 tokens, which limits batch sizes and slows down training and inference.

3.2.2.4 Preprocessing Document Source D (Due Diligence)

Source D contains the smallest and most straightforward documents in our corpus. We do not have full access to Source D because of sensitivity, and instead receive a summary table. The table contains several categories, and with a discrete risk score (an ordinal value) and optional notations (1-2 tokens each). The scores are ordinal variables, and depending on circumstances, some categories may not be present. We preprocess each document simply by flattening the table. Candidates often have multiple documents from Source D, in which case we concatenate the documents and separate them with a special delimiter token.

3.2.2.5 Preprocessing Document Source E (Milestones)

Data Source E comprises a list of milestones that track a candidate’s progress through the entire hiring process, including the public trust evaluation. Source E is not really a document, but rather a time-series of categorical variables. The

milestones are asynchronous in that they are recorded after significant events, instead of at regular intervals.

One of the objectives for our model is to make accurate predictions at any point in the public trust process. We expect that the timing between milestones may carry important information, which the model should take into account. We represent the milestones as a sequence of tokens, where each milestone corresponds to a unique token. We encode the timestamps as a numeric value and append it to the word embedding for the corresponding milestone token. We set the timestamp value for the initial milestone to zero, and each subsequent milestone reflects the number of days elapsed since the first item.

We do not currently use the notes associated with each milestone, as these typically contain information irrelevant to our task, and the notes are usually far more verbose than the milestone text. However, on rare occasions, the notes contain unique data that cannot be found in any of our other sources. We plan to investigate these circumstances further in a future error analysis to determine whether inclusion of the notes data can improve classification accuracy.

3.3 US Patent and Trademark Organization Trademark Case Files

The USPTO publishes this dataset on its public website, providing the following summary of its contents:

The Trademark Case Files Dataset contains detailed information on 12.1 million trademark applications filed with or registrations is-

sued by the USPTO between January 1870 and February 2023. It is derived from the USPTO main database for administering trademarks and includes data on mark characteristics, prosecution events, ownership, classification, third-party oppositions, and renewal history.¹

The trademark dataset offers a partial reconstruction of the data used in the decision process to formally register or renew a trademark in response to an application. The data are a digest of the files that applicants submit to USPTO – those original files are not made available to the public. As the official documentation states in Graham et al. (2013), the trademark case files dataset is ”derived from USPTO administrative records.” Nevertheless, the data that USPTO publishes are already logically grouped into document-like structures that are consistent with how public trust clearance data are organized into discrete documents.

The task that we seek to model for this dataset is predicting whether or not a trademark application results in a successful registration, which is a binary outcome. Each application or case in the trademark data is assigned a unique serial number. Applications can be for new marks or to renew a mark that was previously registered. Renewals are given a new serial number to distinguish them from previous applications. In addition, a single entity – typically a corporation – may have filed many applications and renewals in the dataset.

We have not identified any prior research on predicting outcomes from USPTO’s trademark datasets. The sole empirical study we identified used these to detect

¹<https://www.uspto.gov/ip-policy/economic-research/research-datasets/trademark-case-files-dataset>

disparities in outcomes across various demographic groups (Schuster et al., 2021). This study applied statistical tests to identify disparities and was not focused on using machine learning to predict outcomes. Although the authors applied textual analysis to some of the dataset’s features, their intent was to probabilistically assign gender and ethnicity from applicant names. Although predictive in nature, this subtask did not involve supervised learning due to the lack of ground truth labels, and was therefore much different from our predictive modeling approach. We also do not intend to investigate fairness on the basis of gender or ethnicity because of the inherent unreliability in predicting demographics from names. In addition, many of the trademark applicants are corporations and not individuals, a fact that Schuster et al. (2021) acknowledges. We investigate fairness on a geographic basis to avoid these issues (refer to Section 7.2.2 for details).

3.3.1 Dataset Components

The trademark dataset is split across 15 files, each of which contains a unique set of features. We treat some files as separate document sources and merge some smaller files into larger ones. Most files are unused because they have very limited coverage across dataset elements (present in less than 10% of trademark cases, often much less). We detail the five files that we use in experimentation, followed by brief descriptions of the remaining files. Table 3.3 provides summary statistics on the lengths of the dataset elements in each file, and example contents for each file type are shown in Appendix A.

Table 3.3: Field Length Summary for Trademark Data

Document	Mean	σ	Min	Max
case file	51	0	51	31
owner	57.87	91.87	0	5155
event	46.11	20.92	6	968
statement	1421	1689	1	57848
foreign app	89.12	31.20	0	201

3.3.1.1 case file

The contents of case file are the basic facts for the trademark application, including dates, the type of mark that an entity is seeking to register (e.g., trademark, service mark, certification mark, or collective), characteristics of the mark (text, image, image with text, etc.), details on legal representation, and many other fields. We use 51 fields in case file, so this list is by no means exhaustive.

case file contains only structured data, which is in contrast to most of the document sources for public trust clearances. The majority of the fields are categorical variables, with many being binary indicators. There are some exceptions, including fields containing names, locations, and dates. We do not use all 80 fields, because many represent intermediate or near-final decisions in the application process that correlate strongly with the final outcome, thereby trivializing any prediction task. We eliminate these fields from use through a combination of examining their corre-

lation against the outcome and referencing documentation provided by USPTO.²³

case file also contains the final disposition of the case, which we code as a binary variable for the outcome. There are hundreds of unique values for this field that are essentially variants of three different outcomes – live, dead, and pending. We exclude pending cases and recode the rest to dead (0) or live (1). An example of the partial contents of case file is shown in Appendix A, Figure A.1.

3.3.1.2 event

This file contains the control sequence for each case, and is semi-structured, as the sequences are variable length. Each element in the sequence consists of an event code, an event type, and a date. Event code is a four-letter string, and there are 607 unique codes in the dataset, per Graham et al. (2013). In our sample of the dataset, 590 of the 607 codes are observed. Event type is a single-letter code and appears intended to provide further context, but we exclude it because “event codes of interest tend to have only one possible or an irrelevant event type” (Graham et al., 2013).

Individual event codes each correspond to a single action during the trademark application process, and largely equivalent to milestones in the public trust clearance data. There are far more unique event codes than milestones, and although the control sequences for trademark data are shorter than in public trust data, they often cover much longer date ranges – multiple decades are not unusual. We

²https://www.uspto.gov/sites/default/files/documents/casefiles_schema_high_level_2022_v1.pdf

³https://www.uspto.gov/sites/default/files/documents/varitable_2022_v1.pdf

treat event data exactly the same as milestones in the public trust data – a separate document source consisting of a sequence of textual events that are linked with timestamps. An example of the event data for one case is given in Figure A.2, which contains a relatively short (8 item) event sequence.

3.3.1.3 owner

This source contains data on each owner associated with the application. These data are also semi-structured, due to the variable number of owners for any given record. The owner fields specify the full address of each owner entity, which may be domestic or foreign. There is also a code that indicates the type of entity (individual, corporation, etc), and additional fields that alternate names of the entity, if applicable.

We treat the owner data as its own document source because it is semi-structured and its data are relatively self-contained. We concatenate across fields and then across owner entries to create this document source for model input. As with event, we show a shorter example of the owner data in Figure A.3.

3.3.1.4 statement

This source is primarily unstructured data. Besides the serial number, it contains only two fields: a code that specifies the type of mark across multiple different categories, and a text statement that specifies goods or services associated with the mark. There is no limit to statement length, as some statements are in excess of

55,000 tokens. There can be more than one statement linked to a case. We treat statement as its own document source because its structure and contents differ greatly from the other sources. See Figure A.4 for a partial (truncated) example.

3.3.1.5 foreign app

This source links cases to foreign trademark applications. It is semi-structured data (a case may involve multiple foreign applications) and contains fields on the foreign country, dates for the foreign trademark application and status, whether the mark has been renewed, and the mark’s expiration date. We concatenate data from foreign app with case file because the data types are similar and foreign app does not contain much data. Approximately 30% of our data sample has at least one entry in foreign app. See Figure A.5 for an example.

3.3.1.6 Unused Dataset Contents

classification. This source consists of semi-structured data and relates to the different classes of marks. There are 24 classes in total, which represent broad categories of use. A mark can be associated with a variable number of classes. Each class associated with the mark has a separate entry in these data, which include fields such as dates of first use and status codes for each of the classes. Although we intended to use these data as another document source, the status codes appear to confer information that strongly correlates with outcomes and trivializes the prediction task. As a result, we exclude it from the task.

correspondent_domrep_attorney. This file refers to correspondent address, domestic representative, and attorney. As specified in Graham et al. (2013), this file was separated from case file to reduce file size due to its free-text fields, which refer to the name and address for the correspondent of record for the case, the attorney for the applicant entity, and a domestic representative that is used by some foreign applicants. We exclude this file due to limited coverage across the dataset.

design_search. Specifies the presence of specific design elements (e.g., geometric shapes) for marks that include images. Not included due to relevance and partial overlap with features in case file.

prior_mark Includes data on other marks that the applicant has previously registered. Covers less than 10% of the dataset, and features in case file and event can indicate if the application is a renewal of a prior mark.

madrid_intl_file and and madrid_event. Documents history for international filing. Data only available for approximately 30,000 marks, which is far less than 1% of the dataset.

tm_app_daily. Provides dates of dataset generation, which has no clear relevance to the task.

3.3.2 Preprocessing and Dataset Construction

In keeping with our goal of curating datasets that maintain similarity with the complexity and multi-document nature of the public trust data, we selectively sample individual records from the trademark data. The entire trademark dataset encompasses 12.7 million applications, covering 1870 to the present, according to documentation from the USPTO.⁴ However, the data become increasingly sparse for earlier periods in the collection, as records are incomplete or the era predates the recording of some features. As specified in Graham et al. (2013), applications recorded as abandoned are not visible in the data until 1982.

As noted in table 3.3, there is high variance in the length of the statement and event documents for each data element, and lesser variance for owner, foreign app, and classification. We use data length as a proxy for case complexity, on the basis that cases with more observable features, longer use statements, and a large number of recorded events represent a more challenging decision process. In addition, lengthier data should stress text classification models and afford better insight into which approaches are preferable for a more challenging task.

We sample the dataset by selecting members with the longest token counts, which is consistent with the approach taken by Park et al. (2022) when evaluating transformer models on text classification over long documents. Because the length of statement can range into the tens of thousands of tokens, its length would dominate the token counts and mask potentially useful variance in the lengths of

⁴<https://www.uspto.gov/ip-policy/economic-research/research-datasets/trademark-case-files-dataset>

other features. To compensate, we multiply the length of statement by the sum of the token lengths for all other features, and then take the 60,000 records with the largest values. The application dates for this subset of marks are almost entirely from the 2000-present date range, which is well within the modern era of USPTO record keeping.

We prepare the data for the models using the same processes as for the public trust data. For baseline models, we render all of the features as text and treat each member as a single document. All of our features are categorical or free text, with the exception of a small number of dates; these are binned into 10 categories based on the temporal distance from a common epoch date (January 1, 1980) that predates all records in the data sample. For long documents, we truncate after 1250 tokens, as longer token counts have no effect on accuracy. We order the features so that statement is last, so that truncation only applies to long statements. We also map the alphanumeric code fields in case file into textual descriptions; these mappings are deterministic and based on the explanatory documentation provided by USPTO.

For the ensemble, we segment the data into four separate documents but otherwise treat the data identically. The dates for each observation in event are represented as real values, taking the difference (in years) from January 1, 1980 and appending the value to the embedding for each event. The documents we use for each classifier module in the ensemble are as follows:

- A concatenation of case file and foreign app

- owner
- statement
- event

One key difference with the public trust data is that events in trademark cases are not explicitly connected to the submissions of individual documents. This is partly because we do not have access to the original documents provided by trademark applicants, and also because there is a different methodology for tracking events in the trademark data.

Because of this, we have to take a different approach to using event data in a prediction task. We cannot use the entire event sequence, because the last few items tend to be unique to the binary outcome and would trivialize the task. We instead identify the index in the sequence where the final outcome is evident, and then randomly select a lower index value from a uniform distribution. We truncate the sequence at that point and use the same truncation points in experiments with all models. In randomizing the truncation point, we intend to simulate prediction at all parts of the temporal sequence, up to the point that the actual outcome is self-evident. This approach is in contrast to the public trust data, where we identify a common milestone across all data points and use it for prediction (static task), or select a random nearby milestone in the sequence (dynamic task).

3.4 Congressional Resolutions

Our third dataset is furnished by the US Government Printing Office (GPO), which publishes data on all Congressional bills and resolutions introduced in the US House and Senate from 2003 to the present.⁵ Similarly to the trademark data, this dataset is also a digest and the result of processing that the GPO performed prior to making the data available. We again do not have direct access to the primary source documents; the original full text of bills and resolutions may be available, but they are not directly linked to this data. Other supplementary material could include transcripts of discussions on the House or Senate floor or in individual committees, but if these exist, they are not readily linked to the GPO data. Putting aside these limitations, we are still able to construct a dataset with characteristics that are broadly compatible with the public trust and trademark datasets.

The predictive task for the congressional data is determining whether a House or Senate resolution will successfully pass. Although the GPO's data include bills and resolutions, bills are small in number and very few (less than 5% in an example that we examined) reach a vote and successfully pass. We are using data on simple resolutions from the House and Senate, which number just under 19,000 for the period of 2003 to present. The records are consistent in schema and completeness across the full date range.

The congressional resolution dataset is simpler than the public trust and trade-

⁵<https://www.govinfo.gov/bulkdata/BILLSTATUS>

mark datasets that we introduced, but its inclusion adds useful variance to our dataset pool. Each dataset component contains fewer fields, but is more likely to consist of unstructured text. We again have a control sequence in the form of congressional actions, which track major occurrences in the resolution’s lifecycle (eg, introduction, floor votes, committee referrals, etc). These actions, however, are far fewer in number than in the other datasets, with the average resolution containing only five. This results in a dataset where the control sequence may be significantly less useful; coupled with the simpler feature set, the data may be less amenable to the ensemble approach. We have not identified predictive modeling work on any of the GPO’s Congressional resolution data.

3.4.1 Dataset Components

Each congressional resolution is contained in a single XML file. Because there is only a single, comprehensive document for each dataset member, we will discuss the individual fields that we use as inputs to the model. All of the fields are guaranteed to be present in each XML file, but some may not be populated. In the listing below, we will note the fields that may be left blank. The congressional records frequently contain legislative jargon, and we use official reference materials provided by the GPO and Congress ⁶ to ensure correct interpretation; outside resources we consulted for each field will be given in footnotes. Dataset examples are shown in Appendix A.

⁶<https://www.govinfo.gov/> and <https://www.congress.gov/>

3.4.1.1 Basic Details

Although not a single field, these are a collection of simple details for each resolution, including Congressional chamber, type (resolution, bill, etc.), date introduced, and the numbered Congressional term. These are all structured data, and we concatenate them with some of the other, smaller fields during dataset construction.

3.4.1.2 Actions

This is the control sequence that contains a list of timestamped events. In contrast to the other datasets, all actions frequently fall on the same date, potentially making the temporal information less useful. The date can still contain useful information, since any legislation that has actions pending at the end of a congressional term is rendered dead⁷, so resolutions introduced late in a term may be less likely to pass.

Each individual action consists of a date, the text content of the action (e.g., "Referred to the Committee on Rules and Administration."), and a type, which is a categorical variable. The text content is not as strictly coded as in the trademark data, where the actions are selected from a finite set. Instead, the text follows a predictable pattern but is subject to some variation. Within each action, there may be another data structure specifying a committee related to the action. We do not use this piece of data because it is duplicative of the text content.

⁷<https://ask.loc.gov/law/faq/334496>

3.4.1.3 Sponsors and Cosponsors

Each of these follow the same structure. They contain a list of congressional representatives (full name, party, state, unique ID, and whether they were requested to sponsor or co-sponsor). We flatten the list, concatenate the names and states for each representative, and merge the listings with a separating token for inclusion in the dataset. There is always a single sponsor for each resolution, and some resolutions have no cosponsors.

3.4.1.4 Subjects

This contains a list of legislative topic areas and often includes the names of committees and the policy area of each committee. Each list element is a phrase selected from a finite list of possible phrases. We concatenate the list and use a special token to delimit the elements prior to sending the data to a model.

3.4.1.5 Summary

This contains the official description of the resolution and is typically one to two sentences. If a resolution was amended, there will be a description for each amendment. There may be multiple summaries, each given a timestamp, but they nearly always contain equivalent information. We use the original summary for the resolution and the amendment descriptions, as other data can contain additional text that indicates the final disposition of the resolution.

3.4.1.6 Title

This is another list of free-text elements, containing the official titles of the resolution. Although at times duplicative of the summary, some of the titles can provide additional information. As with similar features, we concatenate the titles.

3.4.1.7 Outcome

Although not an explicit field in the dataset, we determine the binary outcome from the list of actions. The final action in the list (duplicated in the *latestAction* field) usually specifies the disposition. There are some exceptions, such as a star print⁸ being ordered, which is not an outcome for this task but tends to appear after the final disposition.

In order to code the outcomes, we collect a complete list of the possible final actions and construct a list of linguistic patterns that uniquely correspond to each binary outcome. Passed resolutions nearly always contain text about the legislation passing by floor vote or acclimation. Failed resolutions end with a committee referral (and no further actions), a successful motion to table (the key phrase “lay on the table”⁹), or a failed vote on the floor or in a committee.

⁸<https://www.senate.gov/about/research-tools/glossary.htm#S>

⁹<https://www.govinfo.gov/content/pkg/GPO-HPRACTICE-104/pdf/GPO-HPRACTICE-104-30.pdf>

3.4.1.8 Unused Fields

For completeness, we list fields in the XML that are unused. Unused fields tend to be duplicative or have no clear relevance to the task.

committees. A timestamped listing of committees associated with the resolution. These are always listed in *actions*.

policyArea. Duplicative of fields in *subjects*.

textVersions. Contains the first element in *actions* and irrelevant metadata for the XML format.

latestAction. Duplicative of the final element in *actions*.

dublinCore. Boilerplate content that is irrelevant to the prediction task.

3.4.2 Preprocessing and Dataset Construction

We perform no sampling of the congressional resolution data, instead using all 19,000 elements. For baseline models, we concatenate all features and treat each dataset member as a single document. For long documents, we use the same truncation strategy as in the trademark data, as there are no improvements to accuracy when allowing for text over 1500 tokens. We order the features so that sponsor/cosponsor features are last, as these are the longest in the dataset.

For the ensemble, we segment the data into four separate documents but otherwise treat the data identically. The dates for each observation in events are represented as real values, taking the difference (in years) from January 1, 2000 and appending the value to the embeddings. The documents we use for each classifier module in the ensemble are as follows:

- Basic details and title
- Actions
- Sponsors and cosponsors
- Summary

For the list of actions, we repeat the same process as with the trademark data: identifying the action where the final outcome is decided or becomes self-evident, and then selecting a random index to truncate the sequence for prediction. We use the same truncation point for the baselines and the ensemble.

Table 3.4 provides summary statistics on the lengths of each of the fields that we use.

Table 3.4: Field Length Summary for Congressional Resolution Data

Document	Mean	σ	Min	Max
Basic Details	53.03	51.23	2	989
Actions	5.27	3.51	2	31
Sponsor/cosponsors	168.64	291.33	9	4320
Summary	81.92	82.31	0	3253
Titles	28.45	20.72	4	700

4 Model Overview

We choose to model the decision processes in each of the datasets as text classification tasks. We rule out techniques designed for structured data because earlier (unpublished) modeling attempts by stakeholders in the public trust domain did not yield practical results, largely due to information loss in handling semistructured and unstructured features. The trademark dataset contains a similar amount of structured data, whereas the resolution dataset predominantly contains unstructured data; we expect modeling approaches for structured data to be a weak fit for these datasets as well.

Our choice of modeling approaches is driven by the key properties of our three datasets, which contain multiple document sources and a control sequence that journals every event in the decision process. In a typical, single document text classification task, there would be only a single source ($n = 1$) and one outcome. In our tasks, we are instead working with $n = 4$. This raises the question of whether the use of a single model, as in a typical classification task, would be appropriate for the data. The single model formulation would require concatenation of all document sources, which may require truncation for long inputs. There is also potential loss of information from encoding the control sequence into a text-only representation, and the single model approach may be inflexible for an input that changes over time. Robustness to an evolving data source is critical

because we want to generate predictions throughout the decision process. An accurate prediction early in the process is most valuable, as it provides much greater insight to decision makers. Accurate predictions become progressively easier as more data arrive, and near the end of the process become trivial because the final events are often obvious indicators of the outcome. We specify additional goals and constraints based on the nature of the task and real-world operating conditions:

- We want the model to indicate which document sources contribute the most risk and substantially drive the decision.
- The model should be able to train and run inference on constrained resources, primarily a single-GPU system. This requirement reflects the needs of the stakeholders that we described in Chapter 3. We cannot assume that a system owner would have resources for large-scale infrastructure; furthermore, a practical deployment may require many replicated instances to run inference for a userbase.
- The expectation is that the model will be tuned for each of the data sources. This drives our choice of a modular design in the solution architecture.

These criteria motivate our choice of an ensemble, with a dedicated classifier in the ensemble assigned to each data source; a single model design risks data loss and limited responsiveness when handling the control sequence input. Each classifier in the ensemble produces a vector representation for the corresponding data source and uses an MLP to project the combined representation into prediction space. The model will be deployed as an API that takes in the appropriate documents from

each source and generates a label and corresponding likelihood. We hypothesize that an ensemble approach will offer three key advantages over a single model that reads a long, concatenated input document. In Section 5.1 we describe how we will test these hypotheses.

1. The ensemble is better suited to handle heterogeneous data sources. One of our sources is a time series, and treating it as purely unstructured text to match the other data risks a loss of information.
2. The ensemble will be more effective at identifying risk from each document source, which we believe will improve accuracy and facilitate explainability.
3. The ensemble will scale better when presented with longer inputs. This is particularly important for RNN-based architectures, where computational costs increase with longer documents. We also have flexibility to select simpler individual classifiers in environments with limited compute resources.

We train all of the classifiers and the output layers in the ensemble end-to-end. We recognize that some of the classifiers and their corresponding data sources may benefit from pretraining. This may be particularly beneficial for data sources with unique characteristics, such as long document lengths or time series features. We will determine experimentally if data sources benefit from pretraining. A typical pretraining process would involve training a single classifier module against the label set for the overall decision process.

A key requirement for the classification task is to generate accurate predictions throughout the evaluation process. We account for this by truncating the control

sequence at earlier milestones or events in the decision process. It is worth noting that at early steps in the decision process, some document sources will not be available yet. We elect to train multiple variants of the ensemble that are designed to work with fewer sources (i.e., some of the individual classifiers are left out). Although this is cumbersome in comparison to a single model, we do this for two reasons. One is that the model’s performance degrades when one or more inputs are zeroed out. The second reason is that the base probability of a favorable outcome increases as a candidate advances through the process, since candidates are eliminated at each step. We prefer to have separate models that more reliably learn the base probability for a fixed set of document sources, rather than updating the prior each time a new input source becomes available. When presenting our main experimental results in Chapter 5, our comparisons use all data sources in order to represent the maximum complexity of the datasets. We conduct ablation studies in Chapter 6, which reports results when fewer data sources are available.

In the following sections, we discuss the classifiers available for use in the ensemble. Based on the related work survey (Section 2.1), we have chosen two classifiers that we think will perform best on this task. The first uses LSTM layers with specific regularization techniques, and the second uses residual CNN blocks with self-attention, similar to the QANet model from (Yu et al., 2018); to our knowledge, this type of model has not previously been applied to text classification. We select these models to represent contrasting design philosophies. The LSTM classifier is relatively simple, and the extensive use of dropout encourages more parsimony. The original basis for this model was the AWD-LSTM from (Merity

et al., 2017), which achieved state-of-the-art performance as a language model, suggesting that this classifier may be strongly suited to identifying patterns in shorter documents with more freely-written text. The CNN model has a much more complex structure, but the convolution and attention layers are efficient and scale better than an LSTM when handling long inputs. We believe the CNN model will be more effective at finding key patterns in long, form-like documents where the CNN’s translation invariance properties will be useful. For time-series inputs we use a variant of the LSTM model, as our training procedure is designed for use with recurrent layers.

We recognize that large pretrained transformers like BERT and its variants may appear desirable for this task. We are not using these models in the ensemble because the resulting model would be much larger and would violate our constraint of using limited resources. In addition, many document sources are too large for BERT’s limited input size, requiring either truncation or the use of even larger transformer models. These models are still useful as a baseline for comparison, as detailed in Section 5.2.

4.1 Regularized LSTM Classifier

This model reflects a “simpler is better” design philosophy for text classification. It consists of a varying number of LSTM layers (as a tunable parameter, we use 2 based on experimental results), followed by an average pooling layer to create a fixed-size vector representation, and finally an MLP to map the vector to the

output space. When used in the ensemble, we remove the final MLP layer and make use of the document vector representation from the previous layer.

To regularize the LSTM, we implement many of the same techniques used in the AWD-LSTM language model (Merity et al., 2017). The regularization techniques are intended to prevent overfitting in the recurrent connections of the LSTM. These include applying dropout to the LSTM weight matrices (weight-dropping), using dropout on the embedding matrix, and applying a scaled L2 penalty to hidden layer activations at consecutive time steps to discourage large changes in the hidden state. We adapt the model as a classifier by average pooling the sequence of hidden state outputs and feed this to the MLP layer. Adhikari et al. (2019) demonstrated that this simple approach with an LSTM achieved consistently better accuracy than other strongly performing classifiers (excluding BERT) across several well-understood classification datasets.

Many of our data sources contain text that follows a rigid and predictable syntax, a result of the form-based documents they are derived from. We suspect that standard LSTM classifiers may be prone to overfitting and effectively memorizing training inputs, due to their predictable structure, and thereby degrade performance on unseen data. Therefore, the regularization techniques may be particularly useful in improving classification accuracy.

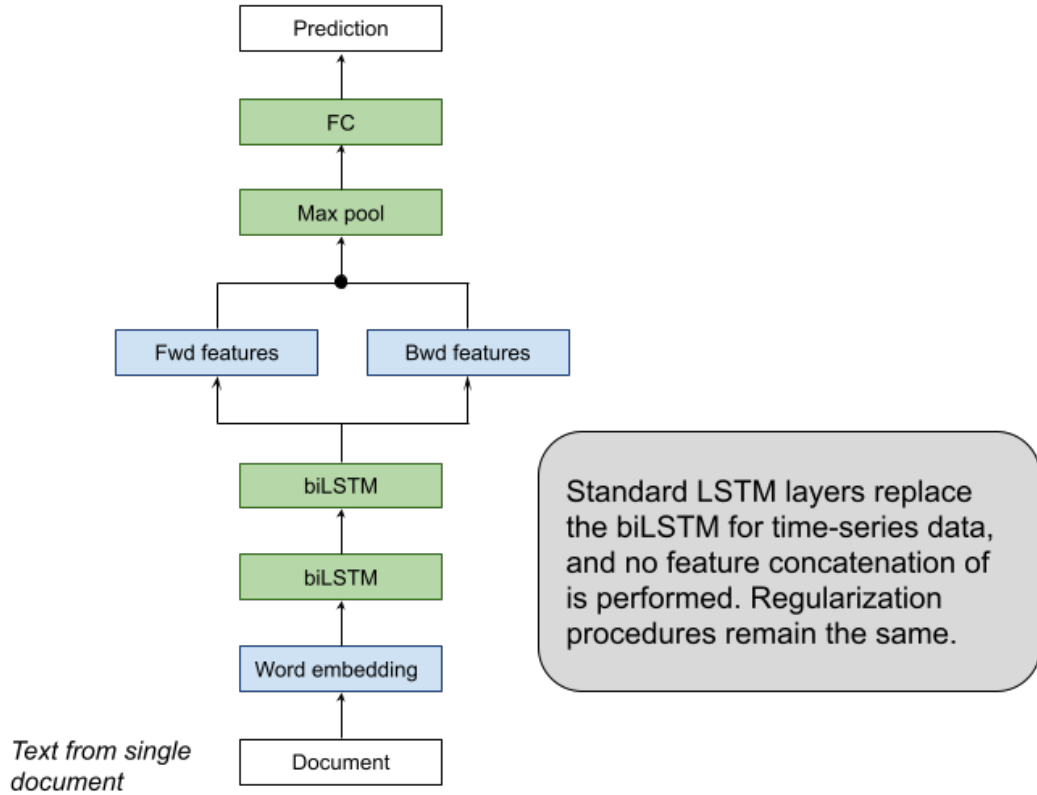


Figure 4.1: Model Diagram for Regularized LSTM

4.2 CNN Model with Self-Attention

In contrast to the LSTM, our proposed CNN model is a much more complicated and expressive model. This model uses techniques from QANet Yu et al. (2018), which was used on question answering and reading comprehension tasks. QANet was later shown in Pugaliya et al. (2019) to outperform a group of models that included BERT on multi-hop QA tasks, where a model has to synthesize related facts scattered throughout a document to arrive at the correct answer. We believe

that multi-hop QA performance is relevant to our task - in documents from Source B, for example, there are often key connections between responses in different document sections that provide important clues to the applicant’s suitability for public trust access. Human evaluators must synthesize these facts to render a judgment, and a model with similar capabilities may be well-suited to the task.

Our implementation uses a similar structure to the encoder portion of the QANet model. The model initially builds a joint word-character representation of the text. Each input token is represented by a word embedding and a sequence of character embeddings; the character sequence is passed through a 1D convolution layer and then a highway network joins the word and character embeddings into a combined representation. The embedding sequences are then passed through multiple stacked residual blocks, as shown below in Figure 4.2. A residual block consists of alternating layernorm and 1D convolution layers (four pairs of these), followed by a layernorm and 8 self-attention heads, and then a layernorm and a fully connected layer. Each pair of layers has a residual connection as well. After four residual blocks, we run max pooling and average pooling over the entire output sequence, concatenate the two resulting vectors, and pass the vector representation through an MLP for the prediction result.

We believe this is the first time that a CNN model with multi-head self-attention had been used in a text classification task. We hypothesize that this model will show better performance than the LSTM on long documents, particularly from Source B in the public trust dataset and the statement document from the trademark dataset.

4.3 Handling Time-Series Data

The control sequence in each dataset consists of a time-series of short text labels (milestones) with corresponding timestamps. The milestones do not occur at regular intervals, and instead mark significant events in the hiring process or document interactions with the applicant. We require a model that can generate predictions at any point in the time series, although the timesteps of practical interest may be much more constrained. In the public trust dataset, for instance, around 40% of the milestones are typically present by the time the evaluation process begins, and by the time 90 – 95% of the sequence is complete, the predictions are usually irrelevant as the outcome is already apparent. We still train over the full sequence because exceptions and unusual circumstances do occur.

To train a model for the control sequence, we adapt the process from Lipton et al. (2015), which proposed an RNN-based approach to predict diagnoses for patients using healthcare data. This approach involved generating a prediction at each time step and then calculating loss over the entire sequence. The same gold output label is assigned to every time step. This results in a network that learns to predict at any time step, and rewards models that predict the correct output early and maintain the correct prediction throughout the sequence. One key difference with our tasks is that control sequence events are not emitted at regular intervals, whereas Lipton et al. (2015) assumed that measurements occurred regularly. We find that by encoding the timestamp as an offset value from the initial milestone and appending it to the embedding vector for each milestone, the model is able to

make accurate predictions. An example of the data and training process is given below in Figure 4.3. Although we experimented with coercing the time series into regular intervals by padding the sequence with blank milestones, we found that this did not improve accuracy. In rare cases where the time series had large gaps between milestones (this can be years or even decades in the trademark data), the padding lengthened the sequence to an extreme degree; we therefore eschewed this practice in favor of parsimonious data.

We use the same regularized LSTM model from Section 4.1 for the time series data, the only modifications being an additional dimension to incorporate the timestamp and the use of a single-direction LSTM instead of a biLSTM. When integrating this model into the ensemble, we are only able to pass a single vector to the final MLP layer. During end-to-end ensemble training, we truncate the time series at a random point and pass the vector output for the final timestep in the sequence. The truncation point is independently chosen for each set of inputs at runtime in order to maximize the variance of the input data. When selecting the truncation point, we only allow selections in the middle 80% of the time series, because at inference time the initial 10% of the time series should always be present, and predictions for the final timesteps are trivially easy.

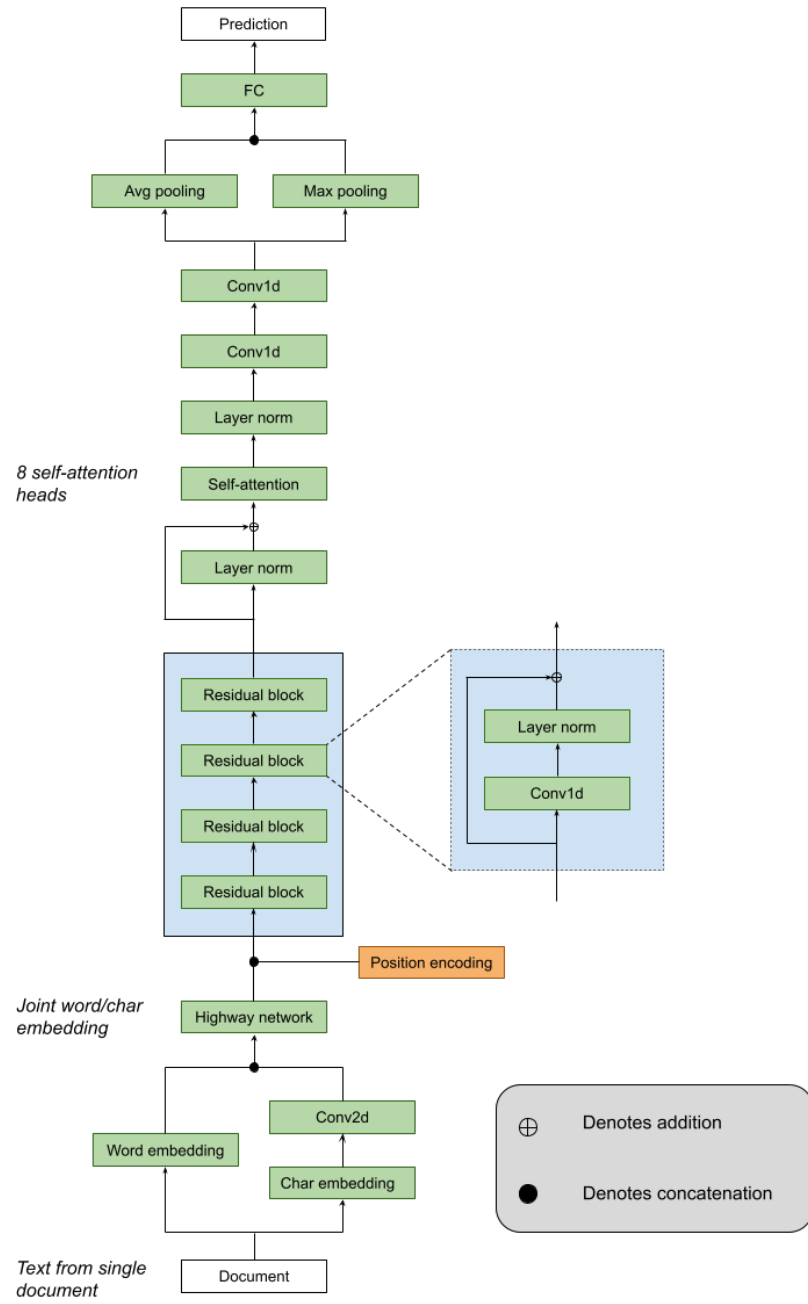


Figure 4.2: Model Diagram for CNN with Self-Attention Module

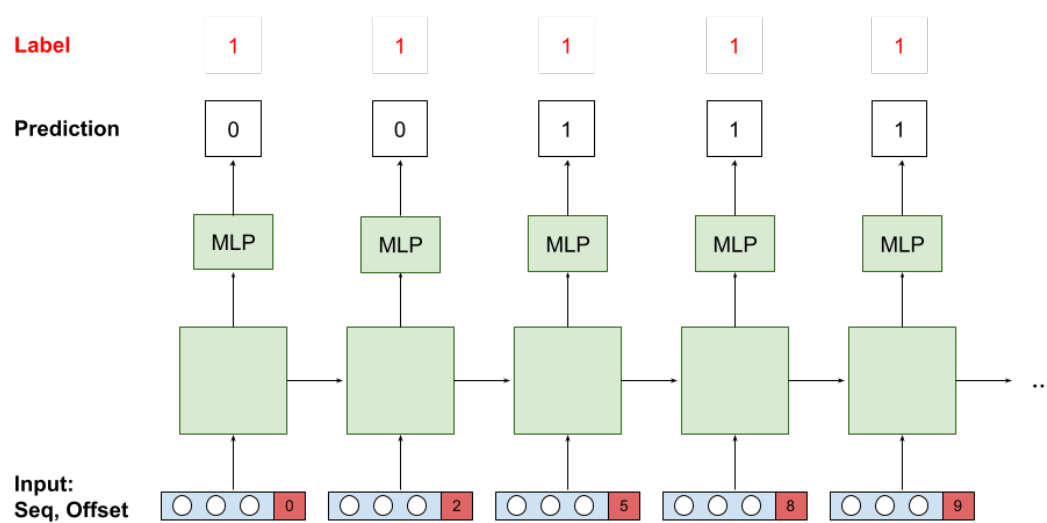


Figure 4.3: Diagram for Predicting Outputs from Time Series Data

5 Experimental Design and Results

Through the experiments we detail in this chapter, we aim to determine whether an ensemble offers a better approach over a single model for multi-document classification. These experiments focus on two of the research hypotheses introduced in Chapter 1, along with additional hypotheses introduced in Chapters 3 and 4:

1. A multisource ensemble of text classifiers will outperform state-of-the-art single source baseline classifiers, and the ensemble can be feasibly implemented with real-world resource constraints.
2. Datasets with greater variety of sources in terms of domain, content (e.g., structured, semi-structured, unstructured data), and length will have a higher performance differential for the ensemble over single source baselines.
3. The ensemble approach will show higher accuracy on dataset members with longer inputs, which are a proxy for greater case complexity.
4. The ensemble will offer higher accuracy when generating predictions at varying timesteps in the decision process, owing to the dedicated classifier module for control sequence data.
5. Performance levels that warrant consideration for real-world use are achievable under the established single GPU resource constraints.

Among stakeholder groups that we identified in the Introduction and in Chapter 3, this set of experiments is primarily focused on end users, who will want to know whether automation or other efficiencies can be realized for their data and tasks. They will also benefit from an understanding of what performance levels can be reached with different model approaches under real-world resource constraints. The results are still of interest to model purveyors, who are likely to be implementing any solution. In order to evaluate the above hypotheses and provide results that will satisfy stakeholder needs, we propose the following tests:

- Compare the accuracy of the ensemble model against multiple static text classifier baselines.
- Implement a testing strategy that emphasizes the dynamic aspects of each dataset, where possible. We expect the model to issue predictions throughout the process, and the testing process should examine the model’s accuracy at various times in the process compared to the baselines.
- Identify dataset members with particularly long inputs and determine if the ensemble approach scales more effectively.
- Identify common error cases in each dataset and determine whether a specific modeling approach is better suited to error-prone inputs.

5.1 Training and Testing Setup

Within each dataset, we determine which individual classifiers perform the best on each individual document source, and then compose the ensemble using the best-performing classifier for each source. We use an 80/10/10 split (train/dev/test) on all datasets, with the dev set used for tuning model hyperparameters. We choose five random seeds and report averages and standard deviations for accuracy and f1 metrics on the test set. Because each of our datasets have slightly imbalanced class labels, we treat f1 score as the more definitive measure when tuning models and discussing results.

Within the context of the ensemble, we pay special attention to the control sequence data. We use the model and training approach specified in Section 4.3. Because this model’s training process differs from the other classifiers, we explore pretraining the classifier on the control sequence for each dataset. The number of pretraining epochs is a tuned parameter. We discuss the pretraining approach in the sections specific to each dataset. No other classifier modules are pretrained, as doing so did not yield improvements to ensemble accuracy.

For all reported tests, we choose five different random seeds and record test set scores for each model. We report averaged results with standard deviations for each model. We select hyperparameters for the ensemble using dev set results only, and reuse the same hyperparameters on the single model LSTM. We report accuracy and f1 scores for binary classification tasks, along with the standard deviations for both metrics across the five seed runs.

5.2 Baselines

We propose four baseline models to offer comparisons against the models we select for inclusion in the ensemble. Efficiency is important to our task because we desire solutions that are feasible in low resource environments, so we include two baselines that do not use deep learning. For these, we use bag of n -gram features where $n \in [1, 3]$ and fit logistic regression and SVM models to generate predictions. We use the scikit-learn (Pedregosa et al., 2011) implementation for both models. We also include a simple multilayer LSTM classifier (hidden layer sizes and layer count tuned on the dev set) and the CNN model with self-attention. We employ the BERT base model (Devlin et al., 2018) as a strong baseline. We expect that BERT’s performance on NLP tasks is likely to be limited by its small context window and out of domain data. However, the model should still provide a worthwhile comparison, given its common use in a broad range of NLP tasks.

For each of the baselines, we treat the input as a concatenation of the document sources. We also treat the control sequence as a static data source, rather than allowing the sequence length to vary during training. This is necessary because all of the neural network baselines fail to converge if we vary the control sequence lengths across training epochs. We address the treatment of the control sequences during training and inference in the dataset-specific sections that follow. We report results on five seeded runs for each baseline, with the exception of logistic regression, as the L-BFGS solver provides a deterministic solution.

We report results on each dataset separately. The discussion of error cases

and key features will focus on the top performing model for each dataset. We also include comparisons between the ensemble and highest performing baseline. Further cross-dataset comparisons are presented in Chapters 6 and 7.

5.3 Results by Dataset

5.3.1 Public Trust Dataset

In addition to the generalized testing procedures above, we specify further testing conditions to better approximate the real-world nature of this dataset. One of our key model requirements is the ability to issue predictions throughout the public trust clearance process. The control sequence (Source E) is the most important data source for this task, as the other sources are relatively static.

For the ensemble, we pretrain the Source E classifier for seven epochs using the process outlined in Section 4.3. We then train the full ensemble end to end, minimizing cross-entropy loss. We concatenate the final hidden layers from each of the document classifiers, and use two fully connected layers to project the concatenated vector to 2-dimensional space for the binary prediction output. This is a departure from the pretraining for the Source E classifier, as now only the final vector in the sequence contributes to the overall prediction.

In order to ensure that the Source E classifier can generate accurate predictions throughout the control sequence, we vary the length of the milestone sequence during training. We perform this randomization process separately during each training epoch. We vary the sequences for the dev and test sets prior to experiments

so that they are consistent across trials. Although we could fix the control sequence to a specific point, such as the milestone corresponding to receipt of the final document source, this would not fully capture the real-world use case. We would also lose insight into the model’s ability to generate predictions throughout the control sequence.

To conduct the randomization, we select an index at which to truncate the sequence. For a set of indices S for a given milestone sequence, we bound the possible truncation index to the interval $S_{trunc} = (0.4 * |S|, |S|)$, where $|S|$ is the length of a given milestone sequence. We randomly select an index from the uniform distribution over S_{trunc} . We bound the index as such because receipt of the final document occurs, on average, at approximately $0.7 * |S|$, which matches the expected value of S_{trunc} . Receipt of the first document, which is the earliest point that our model would begin generating predictions, occurs at around $0.45 * |S|$. With the lower bound at $0.4 * |S|$, we slightly expand the truncation range while keeping $E(S_{trunc})$ at 0.7.

We cannot vary the length of Source E when training baselines, as the models will fail to converge. We truncate the sequence at the milestone that corresponds to receipt of the final document source. The average lengths of the control sequences in the test sets for the ensemble and baselines are both approximately 170, differing by less than one token. Thus, the static and dynamic truncation approaches present the same amount of information to the models.

5.3.1.1 Results on Individual Public Trust Data Sources

We first need to establish which of our classifier modules perform best on each document source so that we can compose an optimal ensemble. We train and test classifiers for data sources A, B, and D, using the outcome of the entire public trust process as the label set. We also include the results for the LSTM on Source E. The accuracy on each individual source is expected to be lower than over the combined data from all sources, as each source presents a limited view of the subject. The results are given in Table 5.1.

Table 5.1: Results for Classifier Modules on Individual Data Sources

Data	Model	Acc (mean)	Acc (σ)	f1 (mean)	f1 (σ)
Source A	LSTM	.7106	.0061	.6538	.0067
	CNN	.7082	.0050	.6493	.0123
Source B	LSTM	.6674	.0066	.6481	.0075
	CNN	.6620	.0065	.6447	.0140
Source D	LSTM	.7146	.0025	.7161	.0058
	CNN	.7142	.0015	.7110	.0020
Source E	LSTM	.7473	.0103	.7358	.0103

The accuracy and f1 scores for each classifier are very close, typically within a single standard deviation. Because the accuracies of the two models are so similar, the set of correctly predicted documents for each classifier differs by around 10-20

elements each, which does not allow for meaningful error analysis. Furthermore, classification from a single data source is a restrictive exercise, as the document in question typically lacks all the information needed to render an accurate prediction. For these reasons, we cannot offer a strong explanation for why one model outperforms the other; in a practical sense, the performance differences are not meaningful. We elect to build the ensemble from regularized LSTM classifiers only because the CNN model’s character-level embeddings substantially increase memory usage.

5.3.1.2 Results on Full Dataset

Table 5.2 shows the results for all models. For the baseline LSTM, we report accuracy on the static test set - where Source E is truncated at the time that the last document is received - as well as the dynamic test set where the truncation point is randomly assigned.

Table 5.2: Ensemble and Single Model Results on Full Dataset

Model	Acc (mean)	Acc (σ)	f1 (mean)	f1 (σ)
Ensemble	.8775	.0055	.8722	.0067
LSTM (dynamic)	.8510	.0050	.8453	.0050
LSTM (static)	.8619	.0025	.8574	.0038
CNN (static)	.8623	.0030	.8587	.0025
BERT	.7503	.1271	.6474	.3265
LR	.7781	-	.7520	-
SVM	.7154	.0011	.6749	.0230

Averaging over five trials, the ensemble outperforms all of the baselines. The LSTM baseline’s performance decays by over a single percentage point if the milestones sequences are varied to match data given to the ensemble, which is better representative of a real-world use case. We note that BERT underperforms the best models by nearly 10 points, partly because the input documents nearly always exceed BERT’s context window. We use the sliding window approach so that BERT sees multiple slices of the full document when the window length is exceeded. The public trust data, being a mix of terse, semi-structured data and unstructured text, is almost certainly out of domain for BERT, which causes further performance degradation. The high variance in BERT’s accuracy scores are due to the model failing to learn the objective on multiple random seeds. On a single run, BERT’s highest accuracy was approximately 0.84, but on other runs it

learned a trivial solution and predicted majority class for every element. Adjusting the number of training epochs did not resolve this issue, but we note that even the best-case runs for BERT did not outperform the LSTM or CNN baselines.

5.3.1.3 Error Analysis

In analyzing the ensemble’s performance, we can identify some clear sources of error based on the semantics of the documents and knowledge of the public trust evaluation process. We begin with a simple confusion matrix for the test set (Table 5.3 to highlight the balance of false positive and false negative errors.

Table 5.3: Confusion Matrix for Ensemble (Left) and Single Model LSTM (Right)

	Ensemble		Single LSTM	
	Label 1	Label 0	Label 1	Label 0
Pred. 1	803	167	776	181
Pred. 0	66	884	93	870

As indicated, the dataset labels are not strictly balanced, with a 45/55 ratio for the hired and not hired outcomes. Both of the top-performing models are prone to predicting the minority class, and incorrectly predicting the hired outcome accounts for 167 of the ensemble’s 233 prediction errors. Although it may appear counterintuitive for the model to prefer the minority class, we find commonalities among members of the dataset where the model incorrectly predicts the hired

case. We provide further results (below) of the model’s accuracy after splitting the not hired outcome into three subgroups. We also include accuracy on the hired category as a comparison (which restates the results from the confusion matrix).

Table 5.4: Ensemble Error Rates by Subgroups of the Outcome Variable

Subgroup	N	Fraction Correct
DQ Type 1	356	.856
DQ Type 2	461	.866
Cancellation	232	.763
Hired	869	.924

The two disqualification types refer to different decision processes in the public trust system. We find that mistakes for both types are most common for subjects whose data appear to be relatively clean - in other words, no major sources of risk in data sources A and B, but the subject is still disqualified. Because our data sources are not comprehensive, there can be key sources of information that influence the actual decision but are not available to the model. This is a more significant issue for DQ Type 1, which partially relies on an additional data source that is unavailable for our work. Another common issue involves fields in Source D that lack specificity, as more detailed information is also not available. This affects the DQ Type 2 cases, and we find that the model’s accuracy is lower when data in these non-specific fields are present.

The model’s performance is the worst on the cancellation group. Members of

this group self-select out of the public trust process before being hired or disqualified. Subjects who canceled after being approved for public trust but before being formally hired are particularly difficult for the model. The model’s accuracy for these cases is slightly above 50%, and this subgroup comprises approximately 14% of the ensemble’s incorrect predictions. One might assume that the source documents for these cases are generally indistinguishable from subjects who are hired. We offer some evidence-based assessments for why the model’s performance on these cases is around 50% and not lower:

- We find that prior to attrition, there may be a significant time period with no milestone activity, or repeated milestones documenting an attempt to reach the subject. These patterns are more likely to appear for subjects who self-select out than for the other outcome subgroups. The Source E classifier’s output consistently shows a sharp decrease in softmax values when these milestones appear.
- In comparison to the overall population and the group with the positive (hired) outcome, the late attrition group is in the evaluation process for longer. The Source E classifier’s output tends to be lower for very long cases, but we cannot control for all of the features that may influence the output.
- Contrary to common intuition, individuals in the opt-out group contain more risk factors than those who are successfully hired, and the model scores these accordingly. However, the 50% of the opt-out group with incorrect predictions are much closer to the hired group, based on manual inspection

of features.

We have access to some of the reasons given when a subject cancels out of the process, and a clear majority of the reasons involve issues that would appear to be exogenous to the public trust process (e.g., accepting a different job, having an emergent family situation, etc.). We leave further exploration of this error source to future work, as additional data sources are likely to be required for the model to better resolve these cases. We provide additional results comparisons in Appendix B (Tables B.1-B.4) for test set cases that cover the self-cancel subgroups, DQ Type 1, and a special case where the applicant provides contradictory information in Sources A and B, which is an indicator of deception. The ensemble’s accuracy is modestly higher for all of these cases.

To evaluate the hypothesis that the ensemble is better suited to handling long inputs than a single model, we sorted the test set by total input length and separated the top decile. We determined this threshold by partitioning the dataset into two groups and selecting the percentile level that maximizes statistical significance between group means. We report confusion matrices for the ensemble and single model LSTM over this subset of the data in Table 5.5. Both models have reduced accuracy on the longer documents (.7604 and .7813, respectively). Document length is considered a proxy for case complexity in this domain, so lower accuracies are to be expected. The single model outperforms the ensemble, which does not support our hypothesis. The difference in accuracy between the modeling approaches is lower compared to the full test set and is non-significant, based on a single-tailed permutation test (100000 iterations), where subgroup membership was randomly

reassigned and a new difference in accuracy means between classifiers was calculated.

Table 5.5: Ensemble and Single Model LSTM Predictions on Long Documents

	Ensemble		Single LSTM	
	Label 1	Label 0	Label 1	Label 0
Pred. 1	85	30	85	26
Pred. 0	16	61	16	65

The ensemble demonstrates higher accuracy than the top-performing single model approach on most test set subgroups that we identified, except for length, although no single subgroup accounts for a substantial portion of the performance margin. We return to this dataset in Chapter 6 to examine dataset characteristics that drive the ensemble’s accuracy advantage.

5.3.2 Trademark Dataset

We replicate the training and testing process detailed for the public trust data, with a few key exceptions. The control sequence for this dataset does not contain timestamped events that are explicitly related to the receipt of documents. We previously used this information to determine a truncation point in the control sequence. Instead, we identify the events that appear toward the end of the control sequence that are closely linked to the binary outcome. Some events contain an

outcome in the description provided in USPTO’s documentation (e.g., DEAD, REGISTERED), and others nearly always appear adjacent to outcome-bearing events. We identify all such events and mark the index, which we refer to t_{cut} . We randomly select a truncation point in the interval $(0.25 \cdot |S|, t_{cut})$, allowing for a wide range of control sequence lengths. We follow this truncation process for the data used in the baseline models and the dev/test sets for the ensemble, using the same truncation points for all models. For the ensemble training data, we randomly select new truncation points during each training epoch.

We institute two minor changes to the control sequence classifier in the ensemble. We do not perform any pretraining, as there is no performance benefit; this is due to the relative simplicity of the control sequence, which is detailed further in Section 6.3. We also transform the control sequence timestamps; rather than providing raw dates to the model, we use January 1, 1980 as a common epoch and subtract this date from the timestamp. We express the difference in years (including fractional values) and provide it to the model. The use of a common epoch accounts for the wide range of starting dates in the control sequence, which can include historical events (stretching back to the early 20th century) that predate the current trademark application. This information contributes to ensemble accuracy.

5.3.2.1 Results on Trademark Data Sources

The CNN classifier shows a slight improvement over the LSTM on the case file data only. This is the only data source so far where the CNN outperforms. During preliminary studies using public trust data for Source C, the CNN also showed higher performance. Both data sources contain a higher frequency of numeric tokens, which suggests that the CNN’s joint representation of character and word embeddings is able to better represent numeric values than the binning strategy we employ for the LSTM. Case file is the only document source across all three datasets where the CNN shows a performance advantage (Source C from the public trust data was not put into use due to limited coverage). In addition, the CNN did not perform as well on the statement data, which has the longest token counts of any data source. As a result, our hypothesis that the CNN model would have an advantage with longer text inputs was incorrect. In accordance with the individual data source findings, we compose the ensemble using the CNN for case file and LSTM classifiers for the other sources.

Table 5.6: Results for Individual Trademark Data Sources

Data	Model	Acc (mean)	Acc (σ)	f1 (mean)	f1 (σ)
Case file	LSTM	.7885	.0047	.8383	.0061
	CNN	.7888	.0013	.8407	.0007
Owner	LSTM	.7861	.0066	.8389	.0075
	CNN	.7717	.0026	.8318	.0022
Statement	LSTM	.7746	.0025	.8246	.0058
	CNN	.7646	.0070	.8172	.0041
Events	LSTM	.6414	.0122	.6753	.0104

5.3.2.2 Results on Full Dataset

The ensemble again demonstrates stronger accuracy than all of the baselines, but by a narrower margin than on the public trust dataset. We explore why the ensemble approach has a greater advantage on the public trust dataset in Chapter 5. The BERT model’s performance is also limited, although it did not learn a trivial, majority-class solution on any of the five seeded runs. We again consider the limited accuracy for BERT the result of the small context window (in particular compared to the statement data source), and the data almost certainly being out of domain. The out of domain nature of the data is more salient for BERT’s lower performance, as the trademark data have larger token counts than the public trust

data, yet the performance gap between BERT and the other models is smaller. In addition, the statement data are unstructured and bear more similarity to BERT’s training data than the other sources in the trademark and public trust data.

Table 5.7: Ensemble and Single Model Results on Full Dataset

Model	Acc (mean)	Acc (σ)	f1 (mean)	f1 (σ)
Ensemble	.9236	.0026	.9394	.0019
LSTM	.9112	.0005	.9286	.0004
CNN	.9018	.0002	.9215	.0018
BERT	.8116	.1106	.8625	.0601
LR	.8636	-	.8879	-
SVM	.8544	.0005	.8797	.0004

Averaging over five trials, the ensemble outperforms all of the baselines. The LSTM baseline’s performance decays by over a single percentage point if the milestones sequences are varied to match data given to the ensemble, which is better representative of a real-world use case. BERT underperforms the best model by more than 10 points, with the limitation of BERT’s context window again being a relevant factor. The metadata-heavy nature of the dataset is also out-of-domain, and although the statement data source consists only of unstructured text, its unusual format (typically a long list of product descriptions) does not match the narrative style text that would be more consistent with BERT’s pretraining corpus. BERT’s accuracy across random seeds shows higher variance than the other

models, but unlike the public trust data, it does not fail to learn the the objective.

5.3.2.3 Error Analysis

Our analysis of the public trust data benefited from direct access to subject matter experts (i.e., the actual decision makers) and the primary source documents used in the decision process. We do not have the same degree of access for the trademark data, with the data sources being a mix of primary source (e.g., the statement data) and case details that have been processed by USPTO. Because of this, much of our discussion will focus on comparing predictions between the single model and the ensemble when controlling for specific characteristics in the input.

One clear difference between the ensemble and best single model is that the ensemble shows a stronger bias toward predicting the outcome for successful registrations, which is the majority class (Table 5.8). These differences remain persist across all seeded runs. Upon sampling 100 cases from the test set for manual comparison, we were unable to identify any patterns or features that drive the ensemble’s preference for predicting the positive label at a higher rate than any of the single model approaches. In order to identify which data sources may contribute this result, we trained a variant of the ensemble (as detailed later in Section 6.1) that generates separate predictions for each individual classifier, and captured values from the softmax layer. Although the variant is less accurate and its predictions are closer to the single model LSTM, it still offers an approximation of the classifiers in the original ensemble. We filter the test set to the elements where the

ensemble and single model disagree, and provide average scores from the individual classifiers in the ensemble. These are reported in Table 5.9, and the scores are split based on whether the ensemble’s prediction was correct. When the ensemble’s prediction is incorrect, the largest upward shift in score is for the statement document, which suggests that the ensemble’s representation of this data source drives some of the differences in prediction.

Table 5.8: Confusion Matrix for Ensemble (Left) and Single Model LSTM (Right)

	Ensemble		Single LSTM	
	Label 1	Label 0	Label 1	Label 0
Pred. 1	3561	412	3465	350
Pred. 0	70	1956	166	2018

Table 5.9: Average Ensemble Classifier Module Softmax Values

Condition	Case file	Owner	Statement	Events
Ensemble Correct	.6007	.6589	.5250	.5664
Ensemble Incorrect	.6036	.6073	.6275	.6096

Next we address the hypothesis that the ensemble performs better than the single model approach when the inputs consist of long documents. Repeating the approach taken with the public trust data, we identified the 45th percentile as the level that maximizes statistical significance when sorting the dataset by length.

We again compare classifier predictions via confusion matrix (Table 5.10). The accuracies over this subset are .9347 for the ensemble and .9305 for the single model. This difference is non-significant, based on a one-tailed permutation test with 100000 iterations.

Table 5.10: Ensemble and Single Model LSTM Predictions on Long Documents

	Ensemble		Single LSTM	
	Label 1	Label 0	Label 1	Label 0
Pred. 1	1949	185	1899	149
Pred. 0	30	1131	80	1167

We did not find the same success in exploring model weakness and error cases as with the public trust data. We had the benefit of supplementary materials that documented the reasoning behind decisions, which was a great source of insight. The trademark dataset is less interpretable, due to its greater metadata content, terse features, and lower availability of original documents. The control sequence contains over 600 unique event codes, which must be mapped to descriptions that are often not very meaningful to the layman. Accordingly, common approaches like sampling error cases and looking for common features or patterns did not yield very useful results. As alternative approach, we split the test set by quadrant of the confusion matrix from Table 5.8 and identified words that were most commonly occurred in each group (but not the others). These are shown in Table 5.11. The model tends to favor feature tokens related to industry (machines, ap-

paratus, industrial) and software (software, downloadable), and disfavoring media (television, entertainment, advertising). Many of these terms reappear when we conduct a similar analysis on this data in Chapter 7, splitting the data by geographic location. Locations with higher predicted success rates, such as Germany, tend to be associated with the same favored words (and vice versa).

Table 5.11: Tokens in Trademark Data Split by Classifier Prediction

TP	TN	FP	FN
software	providing	usa	aids
downloadable	purposes	america	office
nature	services	states	home
namely	advertising	united	suits
machines	entertainment	nosu	academic
field	information	automobiles	metal
data	toy	prev	disease
apparatus	computer	motorcycles	care
parts	paper	crfa	skin
electric	video	cosmetic	trading
industrial	television	hair	applicances
measuring	business	belts	shoes
printing	others	machines	furniture
recorded	pharmaceutical	vehicles	supplies
control	management	skin	structures

The availability of external resources, similar to those available on the public trust data would strengthen our approach for explaining results and identifying areas of weakness in the model. Beyond access to domain expertise, the availability of rubrics and evaluation documents were both important. The evaluation data justified the final decisions and provided explanations that related features to the outcome. These were particularly useful references to consult for incorrect

predictions in order to identify key features and determine if those features were available to the model.

To further underscore the importance of domain expertise for interpreting the trademark dataset, most entities registering a trademark retain specialized legal representation to assist in the process, which also increases the registration success rate (Schuster et al., 2021). Moreover, an internal USPTO investigation from 2020¹ documented shortcomings in examiners’ abilities to accurately identify goods and services from registration applications. This highlights the challenges in data interpretation, as even seasoned experts face similar challenges.

The availability of such resources would warrant revisiting the results for a deeper analysis. We return to this dataset in Chapter 6 to identify characteristics that make the trademark dataset more amenable to the ensemble approach.

5.3.3 Resolution Dataset

The congressional resolution dataset shares a few key similarities to the trademark data, and we take a similar approach to prepare it for use in the models. The control sequence does contain events that map to the receipt of documents, but the sequences are much shorter: the average dataset element contains five events, including one or more events that contain the outcome. We do not allow outcome-bearing events to be included in the input, as they would trivialize the prediction task. We remove these events and select a random index at which to truncate the

¹<https://www.oig.doc.gov/OIGPublications/OIG-21-033-A.pdf>

sequence, with the constraint that the truncated sequence includes at least one event. Similar to the trademark data, we follow same truncation process for all models. We also conduct no pretraining of individual classifiers in the ensemble, as there is no performance benefit.

5.3.3.1 Results on Resolution Data Sources

The LSTM outperforms the CNN in accuracy on all data sources (Table 5.12). Similar to the public trust data, we compose the model from LSTM classifiers only. In the prior datasets, the two data sources where the CNN yielded better accuracy contained many numeric tokens. None of the sources in the congressional resolution data contain many numeric values. The sponsor data source contains some numerics in the district identifiers for US house members (e.g., D-CA-19), but these numbers represent categorical data. Also of note is that the single source accuracies for the summary data are nearly 10 percentage points higher than the other data sources. Perhaps it is not surprising that the success of a congressional resolution can be predicted relatively easily by a short summary of its contents, but the usefulness of a single data source suggests that a simpler model may be more effective than an ensemble of classifiers.

Table 5.12: Results for Individual Resolution Data Sources

Data	Model	Acc (mean)	Acc (σ)	f1 (mean)	f1 (σ)
Basic Details	LSTM	.8123	.0085	.8074	.0059
	CNN	.8090	.0081	.7958	.0083
Sponsor	LSTM	.8176	.0057	.8104	.043
	CNN	.8064	.0047	.8032	.0032
Summary	LSTM	.8912	.0031	.8770	.0024
	CNN	.8869	.0011	.8768	.0028
Actions	LSTM	.8644	.0009	.9009	.0066

5.3.3.2 Results on Full Dataset

In contrast to the other datasets, multiple single model approaches outperform the ensemble. These results establish a general pattern over the three datasets, where the ensemble’s performance advantage decreases with lowered dataset complexity. The congressional resolution dataset is characterized by short control sequences and a greater number of unstructured data sources, although each data source is short. The BERT model’s performance is much closer to the other models, although still not better than the LSTM. The average dataset member is still longer than BERT’s context window. However, the features are much closer to being in-domain for BERT, with most data consisting of multiple sentences or a

list of names.

Table 5.13: Ensemble and Single Model Results on Full Dataset

Model	Acc (mean)	Acc (σ)	f1 (mean)	f1 (σ)
Ensemble	.9261	.0047	.9171	.0022
LSTM	.9366	.0016	.9294	.0018
CNN	.9344	.0020	.9267	.0019
BERT	.9200	.0060	.9091	.0072
LR	.9275	-	.9184	-
SVM	.9190	.0004	.9081	.0004

5.3.3.3 Error Analysis

The congressional resolution data represents a different decision process than the other datasets: rather than a standard set of guidelines that decision makers follow, this is a collective political process whose participants may be sensitive to a wide array of features that are unseen by the model. The identification of error cases is complicated by the fact that we do not have a complete explanation for why each resolution was or was not passed, as well as an accounting of outside factors that may have influenced a decision. Because of this, we follow similar approach to the analysis for the trademark data.

We compare confusion matrices for the ensemble and single model LSTM in Table 5.14. The ensemble’s incorrect predictions tend toward false negatives (ma-

jority class), whereas the single model shows a more balanced error profile. As with the trademark data, we train a separate ensemble that issues predictions for each of the classifier modules. Scores after the softmax layer are recorded in Table 5.15. On test set cases where the ensemble is incorrect, we see a drop in the scores for the sponsor classifier and a larger score increase for the actions classifier.

Table 5.14: Confusion Matrix for Ensemble (Left) and Single Model LSTM (Right)

	Ensemble		Single LSTM	
	Label 1	Label 0	Label 1	Label 0
Pred. 1	768	46	797	60
Pred. 0	79	982	50	968

Table 5.15: Average Ensemble Classifier Module Softmax Values

Condition	Basic Details	Summary	Sponsor	Actions
Ensemble Correct	.4380	.4706	.5634	.2250
Ensemble Incorrect	.4539	.4734	.4761	.4444

For the hypothesis that the ensemble outperforms single model approach for long inputs, we partition at the top decile of the test set, which again maximizes statistical significance. The accuracies over this data subset are .8723 for the ensemble and .9255 for the single model; the accuracies are lower in comparison to the full test set, and the difference between the classifiers has widened by over

5 percentage points (Table 5.16). Repeating the one-tailed permutation test, the difference is significant ($p < .01$), and so contra to the hypothesis, the ensemble performs worse on long documents for this dataset.

Table 5.16: Ensemble and Single Model LSTM Predictions on Long Documents

	Ensemble		Single LSTM	
	Label 1	Label 0	Label 1	Label 0
Pred. 1	87	5	96	4
Pred. 0	19	77	10	78

Similar to the trademark data, we had limited success in determining features that drove model predictions and identifying common sources of error. Using the results from Table 5.15 as a guide, we split the test set by quadrant from the confusion matrix for each model to determine if features or groups of features occurred uniquely within each subgroup of the test set (Table 5.17). The model clearly favored Senate resolutions and disfavored those from the House. Terms related to international affairs (international, nations, Russian) are associated with incorrect predictions, as are names. The presence of names in the false positive group reinforces findings in Table 5.15, in which the sponsor data source may contribute to false positive predictions in the ensemble. Using this information, we examined the error cases in the test set (less than 150 cases total for the best models), but did not find any clear commonalities among resolutions with incorrect predictions. Error cases spanned a wide range of subjects; even among

resolutions that shared a common subject like international affairs, there were no clear similarities.

Table 5.17: Tokens in Resolution Data Split by Classifier Prediction

TP	TN	FP	FN
sen.	rep.	sen.	rep.
committee	1000	committee	michael
measure	hres	rep.	john
senate	representatives	international	international
amended	intro-h	reported	russian
summary	health	house	republic
introduced	sense	united	support
version	expressing	consideration	national
repeated	introduced	human	military
consideration	should	foreign	democratic
rules	support	rule	christopher
bill	introreferral	nations	jim
h.	designation	xiii	joe
cr	expresses	h.	calls
floor	s.	requirement	robert

For an additional analysis, we used the sponsor data to identify which political party most strongly backed each resolution, then use the congressional session feature and historical data to determine if the party was in the minority in the House or Senate. As expected, resolutions backed by the majority party have a higher rate of success (.4801 to .4211). When we compare ensemble to single model accuracy, the ensemble continues to show higher accuracies at a margin that is similar to the overall test set.

Although the congressional resolution data are relatively straightforward to interpret, further error analysis would greatly benefit from subject matter exper-

tise. With specialized knowledge of the public trust process, for instance, we were able to identify cases with non-intuitive outcomes that were associated with much lower prediction accuracy. Similar insights would drive the identification of clear error cases on this dataset. We suspect there are unseen or latent features in the data, such as relationships among representatives or changes in the US political atmosphere that affect outcomes in ways that the models cannot account for.

Overall, we find that the single model LSTM outperforms the ensemble on every subset of the test data that we identified, a clear indicator that the ensemble approach is suboptimal for this dataset. We noted earlier that the resolution dataset is much simpler and has short control sequences in comparison to the other datasets. There is another key difference, which is that the individual data sources strongly correlate with each other. There is overlap between the summary data source and features in basic details (e.g., resolution title), and the political party of the sponsors and cosponsors would naturally be related to the contents of the resolution. The actions reported in the control sequence, such as committee referrals are also connected to the contents of summary. These correlations are quantified in Chapter 6, where we find that data sources with less overlap and correlation are more suitable for the ensemble approach; these findings comport with the results on the resolution data.

5.4 Conclusion

The overall findings across all datasets are highly positive for stakeholders, in particular the end users. Accuracy levels approaching 90% on the public trust data well exceeded minimum levels for practicality cited by domain experts (above 70%). Although we did not have access to domain experts for the other datasets, we observed even higher scores (well above 90% on accuracy and f1) on each. These findings should validate the text classification approach under the resource constraints that we imposed.

The ensemble’s performance was best on datasets with greater complexity, and it outperformed single model baselines (including a state-of-the-art model) on two of the three datasets, which supported all but one of the hypotheses that we advanced at the beginning of this chapter. The ensemble clearly outperformed single model baselines on the public trust data. Performance margins approached three percentage points in accuracy and f1, which constituted error rate reductions in the range of 18% to 20% for each metric. The performance benefit for the ensemble was not as strong on the trademark dataset, where the difference was above one percentage point (error rate reductions of 14% and 15% for accuracy and f1). In contrast, the single model LSTM outperformed the ensemble on the resolution dataset. The public trust and trademark datasets have greater data variety and longer control sequences than the resolution dataset, which is the least complex of the three. The testing process emphasized prediction at varying timesteps throughout the control sequence, and the results supported the hypothesis that the

ensemble would offer better performance across time. The resolution dataset was again the only exception. However, its control sequences consisted of an average of five timesteps prior to truncation, which is short in comparison to the other datasets.

The unsupported hypothesis was that the ensemble would outperform single models on dataset members with long inputs. This hypothesis did not hold for any of the datasets, as ensemble performance was worse on all but the trademark dataset, which had the highest token counts. There was no significant performance change on the public trust data and trademark data, and the single model approach outperformed on long inputs on the resolution dataset by a significant margin. This hypothesis may be worth revisiting on future datasets with very long documents, given the results on the trademark data. Similarly, we expressed an interest in exploring error cases for each dataset to determine if a particular model offered an advantage. We found that the best performing model for each dataset consistently outperformed the others across error cases and other dataset splits.

Collectively, these results convincingly demonstrate that the decision processes represented by each dataset are amenable to automation. Furthermore, a multisource ensemble of classifiers outperforms state-of-the-art single source baseline classifiers, and the ensemble can be feasibly implemented with realistic resource constraints. These are the exact outcomes that stakeholders in the end user group require in order to consider exploring real-world implementations. The results also provide a starting point for model purveyors, who would be implementing any solution. This group is likely to be further interested in understanding what dataset

characteristics motivate the choice of an ensemble, so that they can select the appropriate modeling approach on future, novel datasets. We address this topic in the next chapter.

6 Results Exploration

We demonstrated in Chapter 5 that the ensemble approach to text classification outperformed all single model baselines on the public trust and trademark datasets, but not on the simpler congressional resolution dataset. We now want to determine what dataset characteristics provide an advantage to an ensemble over a single model. This chapter focuses on further exploring each of the datasets to determine what features account for the performance margin between the ensemble and the best-performing single model. Results should inform the model selection process when encountering a new multisource text classification task, which primarily benefits the model purveyor stakeholder group.

The intuition that ensembles should be composed of diverse classifiers is one motivator for our investigation of the datasets. Our focus on disagreement and error diversity among neural classifiers traces back to concepts introduced in Krogh and Vedelsby (1994). Kuncheva and Whitaker (2003) surveyed a number of techniques for measuring ensemble diversity, which in turn drew on ideas established in the development of boosting (Schapire, 1990) and bagging (Breiman, 1996). We apply similar approaches for measuring agreement among classifier modules for each dataset, with the expectation that data sources with greater diversity generate classifiers that are less correlated and compose a stronger ensemble.

We also measure the relative predictive value of each document source within

a dataset. If a single data source accounts for the vast majority of the variance between the positive and negative classes, as measured by classification accuracy using only that source, we expect the ensemble approach to offer less of an advantage. We are interested in identifying cases where a single source dominates the ensemble, and then determining if merging the other sources or shifting to a single model approach is preferable.

Finally, we examine the assumption that the control sequences in each dataset constitute data that are sufficiently out of domain to require a separate classifier module. The public trust and trademark datasets, which have relatively long control sequences, yield higher performance on the ensemble model. This is in contrast to the congressional resolution dataset that contains short control sequences with little to no temporal variance. We are interested in comparing the relative usefulness of the control sequences in each dataset, and determining what features maximize the utility this data source.

6.1 Ensemble Diversity

In testing for diversity within ensembles trained on each dataset, we look to measure how frequently each individual classifier agrees with the others. Our ensemble architecture concatenates the final layer from each classifier module before applying an MLP to generate predictions. In order to approximate each classifier’s predictions, we append another MLP to each individual classifier’s final layer to generate a binary output. This results in a multitask learning objective where the

loss function becomes

$$\mathcal{L} = \mathcal{L}_{ens} + \lambda \cdot \sum_{i=1}^n \mathcal{L}_i$$

where \mathcal{L}_{ens} is the loss for the ensemble and the individual losses for the n classifiers are summed and weighted by λ . We set λ to 0.05 to maintain training stability. We observe no significant penalty (or improvement) to ensemble accuracy with this compound loss, which should indicate that we have not greatly altered the individual classifiers' predictions. The output from each classifier MLP is fed through a softmax, and we report values for the positive label. We use pairwise measures for classifier agreement, starting with the correlation coefficient ρ . This gives us a similarity measure for the softmax values, but we are also interested in how frequently the classifiers disagree in terms of predicted label. We convert softmax values to labels and take the Jaccard similarity, as a simple and straightforward measure of agreement. Table 6.1 reports pairwise measures for each dataset.

An inverse relationship between classifier agreement and ensemble performance holds across the three datasets. The correlation coefficients indicate a weak relationship among the classifiers in the public trust ensemble, a weak-to-moderate effect size for the trademark ensemble, and strong effects for the congressional resolution ensemble. The Jaccard scores show similar effects; the classifiers in the public trust ensemble disagree with each other most of the time, in contrast to the other ensembles where the classifiers agree more often than not.

These results confirm the expectation that ensemble methods perform better when the individual sources in a dataset result in a discordant ensemble of clas-

sifiers. The results also suggest that partitioning features into separate document sources, while minimizing feature correlation across sources, may be an effective strategy to further improve the performance of ensemble methods. We will examine concepts of document partitioning in the next section.

Table 6.1: Ensemble Classifier Agreement by Dataset

Classifier Pair	ρ	Jaccard
Public Trust		
Source A-B	.3083	.3553
Source A-D	.2558	.4504
Source A-E	.3581	.4868
Source B-D	.2377	.3878
Source B-E	.1975	.3170
Source D-E	.1957	.3793
Trademark		
Case File-Owner	.3033	.5943
Case File-Statement	.3341	.5858
Case File-Events	.3420	.6068
Owner-Statement	.3331	.6422
Owner-Events	.5538	.7923
Statement-Events	.2589	.6383
Resolution		
Basic Data-Sponsor	.6519	.6005
Basic Data-Summary	.6910	.6811
Basic Data-Actions	.6706	.5808
Sponsor-Summary	.6706	.6253
Sponsor-Actions	.7127	.6346
Summary-Actions	.6268	.6024

6.2 Ablation Studies

In analyzing the components of the ensemble classifier for each dataset, it is important to understand the relative contributions of each classifier, and by extension, each document source. We perform two different ablation tests to accomplish this. For the first, we ablate the document sources available to the model and observe the changes in accuracy. We are looking for sources that exert a large (or small) influence on the model’s results. Our expectation is that if a single classifier dominates the others, the dataset is probably less suitable for an ensemble approach. We conduct ablations following the ”leave one out” approach, showing results when we subtract each of the data sources while leaving the others intact. We then leave out the least valuable source and ablate the next, until we are left with only a single source.

Table 6.2 contains results from the first ablation test. We do not ablate what we consider the primary document source from each dataset (*Source A*, *Case file*, and *basic details*), as these sources contain the central case facts for each dataset type. One clear result from these tests are that that the control sequences contribute substantial predictive value in all datasets, and this is particularly true for public trust (*Source A*) and trademark (*events*) data.

For the public trust data, *Source B* contributes the least predictive value of the sources, even though it is the largest source by size. This document source substantially overlaps with *Source A*, even though the classifiers for each have only weakly correlated outputs, as established in the previous section. The control

sequence (*Source E*) clearly contributes the most predictive value among ablated sources, as its removal drops accuracy by nearly 13 percentage points (and approx. 6 f1 points). Sources *A* and *D* show similar contributions. These results are consistent with the relative accuracies when training a model on a single data source only (Table 5.6).

On the trademark dataset, we again see that the long, unstructured data in *Statement* is of limited predictive value, contributing less than one point in accuracy and f1. The value of the *Owner* features are also relatively modest, but the *Events* control sequence is substantial. In comparison to the public trust data, accuracy on the trademark data starts at a higher level from a single source, and does not improve as much when additional sources are added.

The results are similar for the congressional resolution data, where the *Sponsor* data, which is also unstructured and the longest source, adds limited value. The *Summary* data source makes the single largest difference of each of the document sources; the reduced value of the control sequence, relative to the other datasets, is due to its short length and lack of variance in temporal data (most events share the same timestamp). These results are consistent with the relative accuracies when training on a single data source (see Table 5.12).

Table 6.2: Ablation Results by Document Source

Document Source	Acc.	f1
Public Trust		
Full ensemble	.8812	.8772
-Source D	.8479	.8411
-Sources B, D	.8365	.8299
-Sources D, E	.7307	.7107
-Sources B, D, E	.7018	.6522
Trademark		
Full ensemble	.9236	.9394
-Statement	.9142	.9317
-Statement, Owner	.8885	.9129
-Statement, Owner, Events	.7920	.8418
Resolution		
Full ensemble	.9366	.9294
-Sponsor	.9253	.9150
-Summary	.8581	.8468
-Sponsor, Summary	.8586	.8460
-Sponsor, Summary, Events	.8128	.7963

For the second test, we want to determine how ensemble accuracy is affected if we merge two or more sources together. Although not ablation in the traditional

sense, this merging exercise allows us to observe effects as the model architecture shifts from an ensemble toward a single model approach. Source merging also enables exploration of whether merging is advantageous for document sources that have similar domains and data formats. In order to avoid exhaustively training models on all possible combinations of merged data sources, we exclude the control sequence and report results from merging all other possible pairs, as well as the triplet of remaining sources. Because merging the control sequence with other sources would incur data loss, we offer a comparison between the results of the single model approach and the merged triplet to demonstrate the effects of merging the control sequence. Table 6.3 contains results from this test.

The merging of pairs of sources in the public trust data results in very small changes to accuracy and f1, regardless of which pair of sources is merged. An approximately one point drop occurs when merging three sources, and then another two points when merging in the control sequence, which produces the single model approach. The fact that a single merge has only small effect, while there is some degradation after two merges, suggests that the individual classifiers may perform best when working with smaller amounts of text. This would account for the degradation when merging three sources, because the pairwise merges have little to no effect regardless of which features are grouped together.

For the trademark data, we see a slight improvement in performance when merging the *case file* and *statement* sources. As previously noted, *statement* is the weakest of the sources in this dataset while also being the longest. The product description given in *statement* can be considered additional context to some de-

scriptive fields in *case file*, so the merging of these data sources does group similar information together, despite the differences in data format. When merging three data sources, we again observe a reduction in accuracy. However, the results are not much different from the single model approach. This result indicates that the treatment of the control sequence as a separate data source may not be as advantageous in the trademark dataset, even though the results in Table 6.2 clearly show that *event* are strongly information bearing for the classification task. We investigate this further in the next section.

The congressional resolution dataset is the only one where the single model approach was superior. The merging of *basic details* and *summary* results in a modest improvement. Both have similar data: *basic details* contains the title of the resolution, which the data in *summary* expands upon. Similar to the trademark data, merging a pair of related document sources again results in a slight decrease in classification error. The merging of three sources produces further (slight) improvement, as the ensemble’s architecture begins to look more like the single model approach.

We find that merging data sources is beneficial when the two sources contain data that are similar or share semantic content. Successful merges also tend to involve data sources that are relatively simple in terms of the variety of features that they contain. Variety in this case is not a substitute for length, as the *statement* source in the trademark data can be quite long but provides a very specific type of information. This is in contrast to the sources in the public trust data, which all contain a large variety of fields.

Table 6.3: Merging Clearance Data Sources

Merged Document Sources	Acc.	f1
Public Trust		
No merge (full ensemble)	.8812	.8772
Sources A + B	.8812	.8746
Sources A + D	.8823	.8745
Sources B + D	.8812	.8768
Sources A + B + D	.8708	.8677
All (single model)	.8510	.8453
Trademark		
No merge	.9236	.9394
Case File + Owner	.9203	.9370
Case File + Statement	.9260	.9401
Statement + Owner	.9038	.9240
Case File + Statement + Owner	.9110	.9302
All (single model)	.9112	.9286
Resolution		
No merge	.9261	.9171
Basic Details + Sponsor	.9189	.9100
Basic Details + Summary	.9285	.9195
Sponsor + Summary	.9216	.9121
Basic Details + Sponsor + Summary	.9285	.9208
All (single model)	.9366	.9294

6.3 Examining Control Sequences

In the previous section, we raised the question of when treating the control sequence as a separate data source is beneficial. We now examine what features in a control sequence motivate using a separate classifier for this data source. We focus on comparisons between the public trust and trademark datasets, as these both contain long control sequences with a high degree of variance among dataset members. The congressional resolution data, in contrast, contains short sequences (five events, on average) that all tend to share the same timestamp. We will offer limited comparisons to the resolution dataset in this section, but the short and simple nature of the resolution dataset’s control sequences will trivialize most of our exploration.

We posit that a control sequence that is most useful to model as a separate data source, rather than a set of static tokens will contain at least some of the following characteristics:

- Each sequence comprises a wide variety of states, and the sequences are relatively long (tens of states or more).
- The initial states in the sequence offers relatively little predictive value toward the final outcome.
- The predictions generated by the classifier over time gradually evolve to favor the correct label. The changes (with respect to a single label) should not be strictly monotonic, as this would imply a trivial process. However, frequent discontinuities are also undesirable.

- As corollary, the sequence should not introduce new information in the final states that substantially change the predicted outcome. Such changes may be the result of an exogenous event that the model cannot account for, which devalues the predictive content of the control sequence.
- The timing of events in the sequence should hold predictive value.

We evaluate the control sequences for the public trust and trademark data to determine how well each conforms to the above criteria. Our expectation is that the public trust data should better conform, as this would account for the comparatively better performance of the ensemble approach. All comparisons are based on classifier models trained on the control sequences for each dataset. We generate predictions at each timestep in the sequence for all elements in the test set, and we use the softmax values for the positive label.

We start by identifying the largest discontinuity in each control sequence, which we define as the greatest difference in softmax values between consecutive predictions. We aggregate these differences across datasets; in Figure 6.1 we show a histogram of the discontinuity values on the left, and the chart on the right shows the locations of discontinuities in the sequence. Because the sequences have varying lengths, we take the index where the discontinuity occurs and normalize it by sequence length.

For the majority of the public trust data (approx. 64%), the largest discontinuity sizes are between 0.1 and 0.35. Nearly 20% of the trademark data control sequences have no discontinuity larger than 0.05, which is a strong indicator that

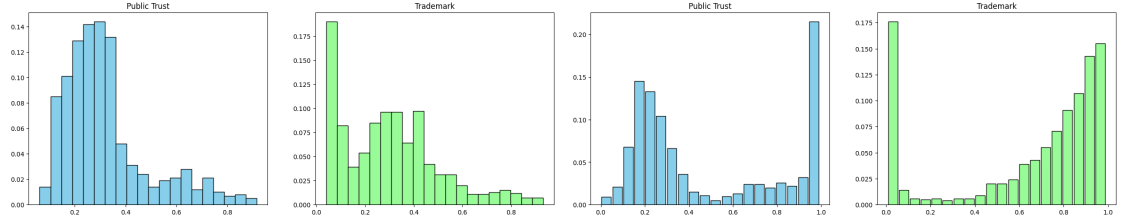


Figure 6.1: Comparing Discontinuity Size (left) and Location (right)

the predictions are consistent throughout the sequence. These sequences appear to represent trivial prediction tasks, as the entire sequence could be replaced with the initial state.

Both datasets show large spike in discontinuities at the end of their sequences. This spike does not necessarily indicate that classifier predictions flipped to the other label. The final elements in the sequences for both datasets tend to be uniquely mapped to a binary outcome, and many of the discontinuities reflect stronger model confidence. There is a concerning trend in the trademark data, as the discontinuity counts rapidly increase in the final third of the sequences. In the public trust data, there is a clear increase in discontinuities around 20% of the way through the sequence. We find that this consistently occurs when an applicant is matched to a job in the hiring process, which signals the start of the public trust clearance process.

We want to resolve whether the discontinuities that frequently occur at the end of the control sequences reflect the model changing the predicted label or growing more confident in the current prediction; the former would violate the criteria that control sequences not introduce critical information at or near the final event. In

Figure 6.2 we reproduce the chart showing where discontinuities occur, but this time filter the data so that only sequences where the initial and final predicted labels differ.

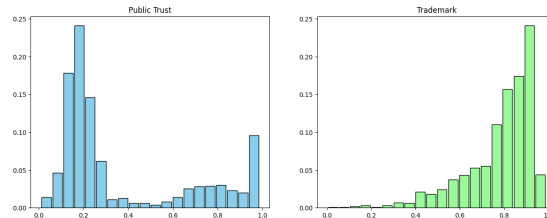


Figure 6.2: Discontinuity Locations When Initial/Final Predictions Disagree

There is a greater contrast between datasets with this filtering in place - the trademark dataset shows a large spike occurring late in the sequence, whereas the spike for the public trust dataset is much smaller. This indicates that when there is at least one change in the predicted label during the sequence, the key evidence in the trademark data tends to arrive late. In the context of our prediction task, the data may arrive too late to be of use to the classifier, which makes the control sequence less useful.

We also compare the predicted labels throughout the control sequences, and Figure 6.3 shows the longest consecutive stretch of predicted labels. A value of 1.0 indicates that the same label was predicted throughout the sequence, and a value of 0 indicates continual alternating predictions. The right side of the figure shows where the changes in label predictions occur. The patterns from the earlier figures for each dataset hold. In over 50% of the trademark data control sequences, the same label is predicted at nearly all timesteps, and when the model

flips predictions, it is more likely to occur later in the sequence. The public trust dataset has fewer sequences dominated by the prediction of the same label and is more likely to show a high level of disagreement (typically softmax scores close to the 0.5 threshold). When predicted labels change, they tend to change in the first quarter of the sequence.

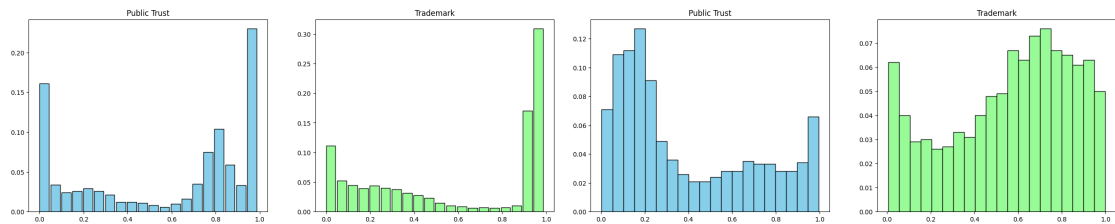


Figure 6.3: Consecutive Predicted Labels (left) and Location of Changes (right)

We find that the public trust data are most consistent with our expectations for control sequences that are amenable to the ensemble approach. Accordingly, this control sequence adds more value to the ensemble than in the other datasets, and there is a greater penalty to accuracy when treating the sequence as a set of static tokens. The reasons for this are clearly shown in the data: the control sequence is not trivially short, the predictions very early in the sequence are not particularly accurate, and there is not a high prevalence of exogenous events late in the sequence that substantially affect predictions.

The ideal characteristics we identified for a control sequence are not necessarily straightforward to identify *a priori* when presented with a new dataset and a similar prediction task. Domain expertise can assist with analyzing control sequence features for suitability with an ensemble of classifiers. A lightweight empirical

approach, by training a classifier on only the control sequence, can also yield results that are straightforward to analyze to determine suitability for the ensemble approach.

6.4 Actionable Results

We find that there are clear, identifiable characteristics common to datasets where the ensemble approach outperforms single model baselines. These characteristics are most pronounced in the public trust data, in which the ensemble holds the greatest performance advantage. This is another positive finding for stakeholders, especially model purveyors, who would be responsible for developing solutions for novel datasets. The results now enable the description of a repeatable process to examine future, unseen datasets for compatibility with the ensemble approach.

Datasets with stronger ensemble accuracy show greater disagreement among classifier modules and higher marginal performance differentials when one or more data sources are ablated. We are able to connect quantitative scores from these tests with qualitative characteristics for each of the data sources. For example, there is limited feature overlap between Sources A and B in the public trust data, but not across the other data sources. In contrast, there is often significant feature duplication across the basic details and summary sources in the resolution data; there are also strong correlations between resolution sponsors and the summary (certain representatives tend to sponsor certain types of legislation), as well as sponsors and actions (specific representatives can exert influence the legislative

process).

Clear patterns arise for the control sequence as well. The control sequences in the resolution data hold less marginal predictive value than in the other datasets. This is the likely result of the control sequences' short length and minimal variance along the time dimension; there is no advantage in representing the control sequence with its own classifier. These characteristics favor the single model approach. The ensemble approach benefits when the control sequences are longer, the initial states offer minimal predictive value, and information that substantially changes the prediction is unlikely to arrive at the very end of the decision process.

Compiling these results, we offer a list of dataset characteristics that indicate a performance advantage for an ensemble. These can save time and resources by identifying an ideal approach without having to train and compare multiple models.

- Individual data sources each offer a view of the data that is largely independent of the other sources.
- Data sources contain little to no feature overlap. When feature overlaps occur, the features have different representations and levels of detail.
- Sources contain a range of data types (structured, semi-structured, unstructured data), including multiple types within a single source.
- Data sources are drawn from different domains, such that developing a common schema would be difficult or unreasonable.

- Control sequences are relatively long, with average lengths on the order of tens or more.
- There is significant variance over the time dimension in the control sequence (i.e., all events do not occur nearly simultaneously).
- Early events in the control sequence are not strongly predictive of the outcome; it is atypical for the predicted outcome to remain the same throughout the entire sequence.
- Critical information that results in a label change or large jump in softmax output may occur throughout the control sequence, not primarily at the end.

Insights from domain expertise may be needed to effectively apply these items to a novel dataset. We leave to future work the development of heuristics that can evaluate an unseen dataset on one or more of the forgoing items at a significantly lower cost than training a model.

7 Examining Fairness in Model Decisions

The experiments and results in this thesis have so far focused on a general classification approach. We now want to outline a basic first analysis for fairness. The results from this analysis are intended to be a starting point for future investigation, as what might initially look like unfairness may not be unfair. Further investigation will require domain expertise for a balanced interpretation of results and determination of next steps.

In earlier discussions about stakeholders, we asserted that end users and model purveyors would hold keen interest in fairness testing. End users will want to understand if a solution systematically disadvantages one or more subgroups, particularly if the solution diverges from the original decision outcomes to treat a given subgroup more unfairly. Decision processes, especially those run at scale by large institutions, often have official documentation stating core values and non-discrimination policies; an unfair solution would violate these principles and therefore be unacceptable to end users. Furthermore, end users would not want to risk operationalizing a solution of unknown fairness, as doing so could expose their institution to lawsuits, penalties, and reputational damage if the solution is unfair.

Model purveyors will be interested in using results from fairness testing to critically examine and potentially refine their decision processes. These refinements

would be broader in scope than remediating an unfair solution. For example, testing may uncover a feature that is potentially important to the decision process but is also a source of outcome discrepancy for a subgroup of concern. Model purveyors may decide to examine this feature and decide whether or not the discrepancy constitutes an unfair bias in the decision process. Such a bias would require remediation, which could include altering, removing, or replacing the feature in question, or even redesigning a part of the decision process.

Our goal in this chapter is to develop a repeatable process for identifying potential sources of unfairness in both the data and the model’s interpretation of the data. We recognize that research on fair classifiers commonly includes work on remediating unfairness. Such studies often focus on datasets such as UCI Adult (Becker and Kohavi, 1996), Propublica’s analysis of COMPAS¹, and medical data (Johnson et al., 2016). However, these datasets tend to comprise structured data and do not involve decision tasks with a considerable degree of subjectivity. As a result, it is relatively straightforward to conclude that discrepancies in predicted outcomes are unfair; this is not the case in our datasets, where discrepancies may arise from group-level traits that correlate strongly with an important feature. Limited exceptions exist, such as Lipton et al. (2018), which includes a case study on graduate CS admissions. However, this study also relies on structured data and omits complex, unstructured sources such as personal statements and letters of recommendation that are important to the decision process. These data sources,

¹<https://www.propublica.org/datastore/dataset/compas-recidivism-risk-score-data-and-analysis>

which are similar to ones found in our datasets, require subjective processes to evaluate. The appearance of discrepancies would raise important questions about whether the decision process was actually unfair, and these questions would need to be answered before taking remediative action.

Before introducing our testing process, we define some terms and notation. Consider a dataset $\{a^n, x^n, y^n\}_{n=1}^N$, where a^n is a sensitive variable or feature (we use the terms interchangeably), x^n is the set of observations or characteristics used to generate a prediction, and $y^n \in \{0, 1\}$ is the binary outcome. Depending on the task, a^n may be present in x^n and used to generate a prediction \hat{y}^n . We represent the real-world decision process as the function d such that $d(x^n) = y^n$. If a model is successful in accurately predicting the outcome of a decision process, then it is identifying correlations among features that are indicative of the actual outcome.

Within the context of our datasets, we primarily consider fairness on the individual level: cases with highly similar features that differ only in terms of one or more sensitive variables should have highly similar predicted outcomes. This definition is consistent with decision guidelines published by agencies that grant various types of clearances²³, as well as USPTO’s manual for processing trademark applications⁴; these sources emphasize individual-level details for decision making and do not mention any guidelines for group-level fairness. We also apply group-level fairness metrics, and assert that a trained solution should not magnify

²³<https://www.dni.gov/files/NCSC/documents/Regulations/SEAD-4-Adjudicative-Guidelines-U.pdf>

³https://www.dcsa.mil/Portals/128/Documents/pv/DODCAF_/DOD_CAF_Whole_Person_Factsheet_042021.pdf

⁴<https://tmep.uspto.gov/RDMS/TMEP/current#/current/TMEP-1200d1e1.html>

cross-group discrepancies in the original decision. Although prior studies (Corbett-Davies et al. (2017), Kleinberg et al. (2016), among others) have established that inherent tradeoffs exist among group and individual fairness measures, both types of fairness are likely to be of interest to stakeholders and should be measured.

For each dataset, we pick at least one sensitive variable to examine for fairness. Our general hypothesis is that there is a bias with respect to each sensitive variable that correlates with the outcome. This bias results in a significant, measurable difference in outcome across subgroups that share specific values of the sensitive variable. The sensitive variables that we examine are all categorical, and most tests involve pairwise comparisons against the dominant subgroup. Because it is impractical to test every sensitive variable in each dataset, we pick salient features that would be of common interest in the domain of each decision process. These choices are independent of the feature selection process that we followed during model development, and they are also pre-registered: we performed no prior fairness tests to inform these choices. The sensitive variables for each dataset are:

- **Public trust:** race, gender
- **Trademark registration:** country of origin
- **Congressional resolutions:** gender of primary sponsor

These choices are consistent with the types of demographic variables examined in previous fairness studies using real-world data, primarily in the criminal justice domain (e.g., COMPAS) as described in Berk (2019) and Kleinberg et al. (2016).

7.1 Ensemble of Tests for Fairness

We propose a series of fairness tests to evaluate a model for each of our three datasets. The first test examines how the inclusion or removal of a sensitive variable a^n in the feature set x^n alters the model’s predictions. We compare classifier results between feature sets that differ in the inclusion of each a^n of concern, examining results for each level of the protected variable along with the baseline rates from the original decision process. For the public trust and congressional datasets, we add the variables of concern to the feature set for the test case; for the trademark data, we remove it. We are interested in whether the changes in the model’s predictions differ among the levels of the protected variable, if the difference is significant, and if the differences across groups are larger in model predictions than in the actual outcomes. We use Pearson’s chi-square test for significance testing when a^n has binary values, and the paired Wilcoxon rank-sum test otherwise. This is a group fairness measure, and we also consider the traditional disparate impact (80%) rule, where the success rate of the minority class should be at least 80% of the majority (or baseline) class:

$$\frac{Pr(\hat{y} = 1 | a^n = i)}{Pr(\hat{y} = 1 | a^n = j)} \leq 0.8$$

In the second test, we train a model to predict the protected variable a^n from the feature set x^n , using the same architecture as the original prediction task. In cases where x^n includes the protected variable, we omit it and other features that would trivialize the task by being a close proxy (we detail all such cases in the

following sections). We intend this test to identify how strongly the feature set correlates with the protected variable. Although we are not measuring a traditional correlation coefficient, text classifiers allow us to estimate the degree of relationship between a textual dataset of arbitrary complexity and a protected variable. In addition, we split the dataset by predicted label and identify words occur most frequently in each group relative to the others. This process will identify stereotypical terms for each group, providing insight into proxies for the protected variable and features that may be driving differences in group outcomes. Krishna et al. (2022) detail a similar process, which they refer to as a protected status model. We consider this test to be an exploratory analysis and do not set thresholds for an acceptable level of correlation between x^n and a given a^n , as this value may be inherent to the decision task. We expect that the results of this test will be particularly useful for future investigations; features that correlate strongly with a disadvantaged subgroup are candidates for evaluation by domain experts to determine if the feature should be excluded from the decision process or down-weighted in the interest of improving fairness.

In the third test, we examine counterfactual fairness. We use the feature set from the first test, and at inference time set the a^n to a specific level. We observe how the model’s predictions change in response. This approach has well-known shortcomings - characteristics like race or gender rarely change in isolation - which can result in unrealistic feature sets. The counterfactual test may appear less relevant to the public trust and congressional data, since the protected variables are not normally included in the features for these datasets. This test still provides

insight into how much variance in the outcome a given a^n can account for in each dataset when we control for all other features. We contend that this is useful information for understanding the decision process and provides starting points for further investigation.

We conduct all tests on the best-performing model for each dataset: we use the ensemble model for public trust and trademark data, and the single-model LSTM for the congressional resolutions.

7.2 Fairness Evaluations by Dataset

7.2.1 Fairness Evaluation on Public Trust Dataset

For the public trust data, we consider the protected variables of race and gender and test to determine if there are disparities in outcome across race and gender groups. It is worth mentioning that during the feature selection process, we removed many features that probably correlate with one or more protected variables, because the features had no identifiable predictive value. These features include street-level addresses, zip codes, and some occupational details. Our fairness tests should help illustrate whether or not the omission of these features results in a more fair classifier relative to base outcomes.

We show the levels of the variables for race and ethnicity in the test set below. For race, we omit the *not reported* group from test 2 due to small sample sizes. There are four additional levels not shown here because their sample sizes are under 10; we omit these from the analyses. For gender, we focus on male and female

values; there is also a *not reported* group with four members that we exclude. The size of the test set remains 1920, which includes all the groups omitted from Table 7.1.

Table 7.1: Race and Gender Breakdown in Test Set

Race	N	Gender	N
White	1297	Male	1127
Black	217	Female	789
Asian	176		
Hispanic	166		
Not reported	50		

Table 7.2 shows the base rate of applicants in each group who were granted a clearance, alongside the number of applicants for whom the model predicted to clear. The model predicts the *clearance granted* outcome at a higher rate than the true outcomes for all groups except for the *not reported* and *female* groups. With the exception of the *not reported* group (which is not a race), the model’s predictions show less inter-group variance than the actual outcomes, which suggests that the combination of feature selection and omission of the protected variables is successful in reducing discrepancies across groups.

We use the *white* and *male* groups as comparison baselines, since they represent the majority groups. The only group that fails to meet the 80% test for disparate impact is the model’s *not reported* race group, and this difference is significant.

Differences in rates, relative to baseline, for the other groups are all non-significant. The *not reported* group holds a unique property, as the subject has elected to withhold information. In discussions with domain experts, it is widely believed that applicants who are less cooperative in providing information tend to be less successful in the process, and this is reflected in the base rates. The classifier magnifies this effect, which may at least partially be the result of the *not reported* group constituting less than 3% of the dataset.

Table 7.2: True and Predicted Clearance Rates by Race and Gender

Race	Cleared	Predicted	Gender	Cleared	Predicted
White	.490	.504	Male	.426	.519
Black	.433	.548	Female	.516	.487
Asian	.405	.468			
Hispanic	.427	.484			
Not reported	.400	.100*			

We now explicitly include race and gender in the input features and collect model results. We add the features to data sources A and B (as described in Section 3.2), which is consistent with where these data occur in the original sources. Including these features yields a slight decrease in accuracy compared to the original feature set; this was a typical outcome during feature selection when including features with no additional predictive value.

Table 7.3: Classifier Accuracy after Including Race/Gender Features

Feature Set	Accuracy	f1
Original	.8820	.8787
+ Race	.8796	.8745
+ Gender	.8792	.8465
+ Both	.8807	.8771

We also compare the predicted clearance rates for the original feature set and the modified features with race and gender included. The predicted clearance rates for most demographics shift in the direction of the base rates reported in Table 7.2. The model becomes more discriminatory than the base rates for most of the groups; the *black*, *Asian*, and *not reported* groups all violate the 80% test for disparate impact, and the differences are significant for each of these groups and for the *female* group.

Simply by blinding the model to race and gender features, coupled with pruning via feature selection, we observe greater parity across groups in the model's predictions. The only exception is the group of applicants who decline to provide race data, where the model imposes a strong penalty for withholding such data.

Table 7.4: Predicted Clearance Rates with Race/Gender Added to Features

Race	Orig. Added		Gender	Orig.	Added
White	.504	.484	Male	.519	.403
Black	.548	.354*	Female	.487	.523*
Asian	.468	.380*			
Hispanic	.484	.458			
Not reported	.100*	.275*			

For the second fairness test, we train the model to predict race and gender as outcome variables. We combine categories omitted from Table 7.1 into a single level rather than remove elements from the dataset. We find a test set accuracy of **0.8421** for predicting race and **0.7870** for predicting gender. These show that the feature set strongly correlates with both demographic categories. Of the four data sources in the public trust dataset, only source A and source B exhibit correlations with race or gender. When training on a single data source, the model fails to learn the objective when using data sources D and E; it learns a trivial solution of predicting the majority class, which indicates that both sources are effectively uncorrelated with the protected variables. Sources A and B, in contrast, contain biographical features that we would expect to correlate with race and gender.

In order to better illustrate these correlations, we split the dataset elements by predicted label and identify feature tokens with the largest differences in occurrence between a target label group and the other groups. We provide a summary of the

types of tokens that are most closely associated with each label in Tables 7.5 and 7.6 (actual tokens are not provided due to proprietary restrictions on the dataset). We do not provide tokens associated with the *not reported* group because the model fails to learn a decision boundary for this label, generating very few predictions for it.

One consistent pattern is that demographic categories with lower clearance rates tend to have more tokens associated with negative information (financial issues, alcohol use, etc). This is a likely explanation for why the *black* and *male* groups have lower clearance rates. The most common tokens in the *Asian* and *hispanic* groups relate to overseas travel and experience. Although not necessarily negative factors in the clearance domain, published government adjudication guidelines like the SEAD-4 evaluate factors like foreign preference, which probably account for the lower rates for these groups. The cross-group differences in predicted clearance rates increase when the model has access to race and gender as part of the input, which suggests that the model may penalize these demographic groups when stereotypical negative features appear in the input. We examine this as part of the third test.

Table 7.5: Summary of Tokens that Co-Occur with Racial Group

White	Asian	Black	Hispanic
US birth	Travel to Asia	Maryland	Latin America
Military exp.	Asian languages	Financial issues	Florida
Virginia	STEM degrees	Caribbean travel	Associate degree
European travel	Internships	HR experience	Telecom experience
Bachelor degree	California	Layoffs/work issues	Romance languages

Table 7.6: Summary of Tokens that Co-Occur with Gender Group

Male	Female
Military exp.	Travel to Europe
Alcohol use	Education
Travel to Asia	Humanities
Drug use	Foreign language
Layoffs/work issues	Financial issues

For the counterfactual fairness test, we again include race and gender as input features. At inference time, we change the demographic feature to a specific level and measure the response in the model’s output, both in terms of softmax score and the predicted label.

Table 7.7 summarizes the results from counterfactual testing. Values reported

under the **N** column are the number of test set members for which race/gender were modified (i.e., the group’s population subtracted from 1920). **Avg Chg.** reports how much the softmax score moved for each category, and the next two columns count how many members had increased or decreased scores. The final two columns report how often the predicted label shifted to 1 or 0.

The results for counterfactual fairness are generally follow the results for group fairness. A small number of dataset elements within each group experience large shifts in softmax scores, typically negative and with magnitudes above 0.5. These occur most frequently when the dataset member contains features that are more commonly associated with the target race or gender group, which indicates interaction effects between the protected variables and other features. The *Asian* and *not reported* groups experience a statistically significant difference in softmax compared to the majority group.

From the three fairness tests, we conclude that the model and feature set are generally more equitable with respect to race and gender than the original decision process. We use the paired Wilcoxon rank-sum test to determine significance for the softmax values, comparing scores for each subgroup to baseline categories (white for race, male for gender). For the significance test, we include only the dataset elements that were originally in the baseline subgroup (e.g., male candidates that shifted to female during the counterfactual test). All subgroups show a significant difference in score from the baselines ($p < 0.01$) with Bonferroni corrections applied. This raises the question of whether the effect sizes hold practical significance. We can compute an effect size from the Wilcoxon test, and according

to Cohen’s (Cohen, 1998) effect size classifications, most groups show a strong effect (approx. 0.5 or above), with the exception of the black group (approx. 0.19). However, these figures do not provide explicit practicality measures in the context of our particular task, and no such guidance for practicality yet exists. We note that only two groups (Asian and not reported) have scores that differ by more than one percentage point, and both have a much greater count of the predicted labels shifting to the negative prediction (45 and 119) than the other groups. This suggests that both groups experience a practical level of discrimination when race is included as an explicit feature in the data.

Further investigation is needed to determine whether differences across groups are discriminatory, and the Asian group should be a clear priority for additional study. The large penalty assessed toward the not reported group, even in the absence of racial features, is also notable. Due to the small size of this group, a manual examination of features may be warranted to determine why the model treats this group differently; there may be additional features that are absent or unreported that causes this group to be an outlier.

Table 7.7: Counterfactual Fairness Results for Public Trust Data

Group	N	Avg Chg.	Score \uparrow	Score \downarrow	To 1	To 0
White	623	.0110	474	149	8	0
Black	1703	.0038*	900	803	11	2
Asian	1744	-.0215*	53	1691	2	45
Hispanic	1754	-.0012*	343	1411	6	11
NR (race)	1869	-.0544*	72	1797	0	119
Male	793	-.0069	4	789	0	8
Female	1131	.0072*	1129	2	8	0

7.2.2 Fairness Evaluation on Trademark Dataset

For the trademark registration dataset, we examine the primary location of the owning entity (individual or company/institution) that submitted the trademark. When dataset elements contain multiple locations for an entity, we choose the location that occurs the most often, and in the event of a tie, pick the one listed first. We code location by US state for domestic entities and country for international owners. We note that a previous study (Schuster et al., 2021) documented an extensive investigation into gender and race disparity in trademark prosecutions, which we do not elect to reproduce. Their work focused on applications filed by individuals who were US citizens, but such requirements would eliminate applications by institutions and foreign citizens, which constitute the majority of

our dataset. Furthermore, Schuster et al. (2021) estimates race and gender using probabilities derived from US Census data, which introduces coding errors that we wish to avoid.

Within our sample of 50,000 elements, there are nearly 200 unique locations available; we focus on the top 11 locations plus twelfth catch-all group for the remainders. The top 11 locations account for 68.6% of the test set, with each group comprising at least 2% of the test set. If we considered additional groups, the sample sizes would be at a similar or smaller proportion as the *not reported* group from Section 7.2.1, which was too small for the second fairness test.

The feature set for the trademark data already includes the full address of the submitting entity (i.e., the a^n for this dataset). This is in contrast to the clearance data, where we excluded race and gender because they are protected categories that should be exogenous to the decision process. Within the trademark domain, location is not a protected category, and it holds predictive value. For the first fairness test, we remove location data from the feature set and compare the model’s predicted registration rates by location group. Although the feature set still includes the name of the entity applying for trademark registration, entity name does not perfectly correlate with location, as a single entity can be coded for different locations across different trademark applications.

Table 7.8 lists the 12 location groups. Reported values are for the original feature set, the feature set with location data removed, and the base rates from the actual outcomes. Significance testing for the model results uses the *other* group as a baseline, and more than half of the groups show a significant difference.

Applications from entities in California fail the 80% test for disparate impact, and this result holds for the model predictions and the actual outcomes.

The removal of location data from the feature set results in minimal changes in the predicted rates, and the largest observed difference (for France) is 1.6 percentage points. We also compare classifier accuracy when removing location data (Table 7.9). There is predictive value in location data, even though these features do not significantly change the model’s prediction rates for each location group.

Table 7.8: Location Groups and Registration Rates in Test Set

Location	N	Pred.	Pred (no loc.)	Actual
UK	819	.714*	.722*	.652
Delaware	751	.607	.603	.562
Germany	719	.701*	.695*	.643
France	378	.624*	.608*	.569
Japan	325	.652	.640	.606
California	222	.409*	.401*	.360
Italy	219	.721*	.712*	.658
Canada	196	.739*	.745*	.688
Switzerland	190	.716*	.700*	.689
Australia	174	.752*	.747*	.678
New York	123	.552	.545	.528
Other	1883	.640	.640	.599

Table 7.9: Classifier Accuracy after Removing Location Data

Feature Set	Accuracy	f1
Original	.9207	.9368
- Location	.9055	.9243

For the second test, we remove all location features and train the ensemble model to predict the location groups from Table 7.8. We observe a test set accuracy of **0.8500**, similar to the results for predicting race in the public trust data. This is a clear indicator that the input features correlate strongly with location. We also identify tokens that are most unique to each location group (with the exception of the *Other* group), shown in Table 7.10. Many of these tokens are foreign language words or company terms that are specific to a single country (société, GMBH, S.p.A., etc.). The majority of these tokens occur in the goods and services document and in the document that contains information on the submitting entity. Notably, when we ablated each of the document sources in this dataset, the goods and services document contributed less predictive information than the other sources; its removal resulted in less than a point of accuracy loss. This indicates that many of the features that correlate most strongly with location do not strongly influence the model’s predictions.

For the test on counterfactual fairness, we focus on the top 11 countries only, since the catch-all *other* group is not a specific location. We remove finely-grained location features (city, state, street address) that would act as a proxy to the

Table 7.10: Summary of Tokens that Co-Occur with Location Group

UK	Delaware	Germany	France
limited	software	machines	société
services exp.	health	gmbh issues	data
computer	information	apparatus	advertising
games	computer	parts	transmission
software	management	electric	cards
information	data	vehicles	access
downloadable	llc	metal	management
video	eext	devices	anonyme
entertainment	online	electrical	telecommunication
music	care	control	radio
Japan	California	Italy	Canada
machines	toy	bags	cnsI
apparatus exp.	leather	gmbh	cnsi
maintenance	bags	metal	gnsI
rental	television	s.p.a.	gnsi
electrical	audio	clothing	hemp
retail	video	shoes	health
wholesale	game	precious	financial
industrial	children	creams	ltd.
power	educational	jackets	phones
kabusiki	music	s.r.l.	bath
Switzerland	Australia	New York	Other
sports	services	services	services
ag exp.	ltd	advertising	entertainment
systems	pty	business	business
financial	music	educational	educational
bags	nature	hair	online
electronic	real estate	entertainment	providing
transport	financial	sports	field
control	marketing	events	electric
sensors	educational	marketing	conducting
machines	chocolate	computer	pharmacy

location variable. A complete listing of results is shown in Table 7.11.

We again use the paired Wilcoxon rank-sum test for significance, and the *other* group serves as a baseline for comparison. All of the score changes are significant ($p < 0.01$). Effect sizes are mostly in the weak to moderate range, with only Italy, Japan, and Delaware showing strong effects ($r > 0.5$). In terms of practicality, all of the average score changes are below one percentage point. The locations with the strongest effect sizes paradoxically have more predictions shift to the positive label than the negative. This is because many of the largest negative score changes occur on dataset elements where the model is strongly predicting the positive outcome, but the score decrease does not cross the 0.5 threshold. As with the public trust data, there are no existing guidelines for effect size practicality. Given the results, we can be confident that there are systematic differences among the groups, but the size of the differences appear to effect only a minor change in the predictions.

Unlike the other datasets, most of the changes have a negative sign, even for locations that the model tends to favor. A possible reason is that the counterfactual cases have unusual feature combinations that result in lower scores. When examining individual dataset elements, we find score changes to be modest; in contrast to the public trust dataset, there are very few data points with large score swings (0.5 points or more). This is despite similar accuracy scores for both datasets when predicting the sensitive attribute. One difference between the two datasets is that domain expertise informed the feature selection process for the public trust data, and we selected out many features with no measurable predictive value that were highly likely to correlate with sensitive attributes. Adding the sensitive attributes

to the feature set resulted in large shifts in model predictions across groups. No such shift occurred for the trademark data, and the slight change in scores in the counterfactual test may be the result of more proxies in the features.

Table 7.11: Counterfactual Fairness Results for Trademark Data

Group	N	Avg Chg.	Score \uparrow	Score \downarrow	To 1	To 0
UK	5180	-.0086*	1825	3348	18	14
Delaware	5248	-.0068*	1397	3845	41	15
Germany	5280	-.0009*	2619	2653	9	18
France	5621	.0038*	3124	2490	12	19
Japan	5674	-.0062*	1187	4478	33	6
California	5777	-.0076*	1865	3901	5	66
Italy	5780	-.0072*	1040	4736	36	5
Canada	5803	-.0017*	2709	3038	16	18
Switzerland	5809	.0020*	3211	2568	7	21
Australia	5825	-.0039*	1738	4081	20	14
New York	5876	-2.325×10^{-5} *	2858	3016	19	37

We conclude that the trademark dataset and ensemble model favor some locations over others, in accordance with the base rates from actual trademark application outcomes. There are non-location features that correlate strongly with the submitting entity’s location, but these features appear to exert a modest influence on model predictions. It is less clear whether the differences in success rates among

the locations we examined are the result of a bias.

We note that the USPTO is known to be more stringent with foreign patent applications⁵. This suggests that foreign entities would have lower success rates for trademark applications, which is the opposite of the results in our dataset. A possible reason is that foreign applicants in our sample tend to be mature companies that have higher success rates, which may be a side effect of our sampling process that selected for complex cases. We note that the entities that occur most often in this dataset are luxury goods firms, such as Burberry and Gucci, that are mostly foreign. These firms would be expected to aggressively defend and renew their trademarks, but they cannot account for all of the variance in success rates across location groups. Also, USPTO requires foreign applicants to file with a US-based attorney, and a previous analysis of trademark cases (Gerhardt and Lee, 2022) indicated that entities filing with an attorney have higher registration rates than those who file pro se.

More work is needed to investigate why success rates vary. As we discussed when introducing this dataset in Chapter 4, the trademark case data consist of a large number of metadata fields, which makes interpretation more challenging. Our findings can inform a more detailed evaluation of the data, which would particularly benefit from the inclusion of primary source documents that USPTO received as part of the trademark application.

⁵Some examples can be seen at <https://www.uspto.gov/sites/default/files/documents/PracticeTip-ForeignEntity2d.pdf>

7.2.3 Congressional Resolution Dataset

For the congressional resolution data, we again examine gender as a protected category. We focus on the gender of the resolution’s sponsor, noting that each resolution has one designated sponsor and any number of cosponsors. Cosponsors can number into the tens or even hundreds (for the US House), so the gender of the resolution’s sponsor offers a single variable to isolate and test.

We treat the list of sponsor and cosponsors as a self-contained document source in the congressional resolution dataset. Gender is not explicitly given in the source data, and we do not include it in the feature set. However, the data and features include the full names of House and Senate representatives, along with their district and state identifiers (e.g., *D-PA*). Names and identifiers are proxies for gender, although identifiers require a date or congressional session number (both included in the features). We control for these relationships among features when preparing the dataset for fairness tests. Because gender is not explicitly coded in our source data, we cross-reference additional data⁶ to avoid errors.

In the first fairness test, we examine changes to model predictions and accuracy when we explicitly include sponsor and cosponsor gender as an input feature. Table 7.12 compares rates of successful resolutions in the male and female groups. The first column gives the rates from the actual outcomes, and the next columns provide the predicted rates using the original feature set and a modified feature set that includes gender. The differences across the original and modified feature

⁶<https://cawpdata.rutgers.edu/women-elected-officials>

set and the actual outcomes are not significant. The differences across the male and female groups are significant, but they do not cross the 80% threshold for disparate impact.

Table 7.12: Actual and Predicted Success Rates for Resolutions

Sponsor	N	Actual	Pred	Pred. w/Gender
Male	1448	.466	.460*	.447*
Female	427	.403	.393*	.396*

We also compare changes in model accuracy when including gender as a feature. Because name is a clear proxy for gender, we train and test models with feature sets that include name, gender, and both, in order to control for proxy effects (the original feature set includes names but not gender). The inclusion of gender features does not improve prediction accuracy, which indicates that gender alone does not contribute unique information. When we remove names and do not include gender features, such that the sponsor/cosponsor source consists solely of state or district identifiers, there is a similar decrease in accuracy.

Table 7.13: Classifier Accuracy after Including Race/Gender Features

Feature Set	Accuracy	f1
Original	.9408	.9344
+ Gender	.9232	.9137
+ Gender, no names	.9301	.9212
No gender or names	.9285	.9204

For the subsequent test, we predict gender as the outcome variable. We remove the names of sponsors and cosponsors, as otherwise the proxy effect from names results in a trivially easy task (the model records near-perfect accuracy). We observe prediction accuracy of **0.9456** and an f1 score of **0.8770**, with the lower f1 score resulting from imbalanced labels. This result is a clear indicator that the feature set strongly correlates with sponsor gender. We identify feature tokens that uniquely occur in each group in Table 7.14. There is a clear separation in the token sets for each group, as tokens associated with women, children, and families occur much more frequently in the female group. The three district identifiers that occur in the female group, when cross-referenced with the congressional sessions where these identifiers appear, correspond to female members of the US House. Resolutions that include tokens from the female group have lower success rates than resolutions that include tokens from the male group, which further reinforces inter-group outcome differences.

For the counterfactual test, we include gender in the feature set and adjust it

Table 7.14: Tokens Uniquely Appearing by Gender

Male	Female
house	woman
senate	month
congress	health
international	national
foreign	goals
representatives	awareness
president	children
summary	education
anniversary	families
state	support
federal	history
actions	commemorative
rights	d-ca-13
human	d-tx-18
appropriations	d-mi-13

at inference time. We also remove representative names, which again may obscure effects of gender features due to being a proxy. The results in Table 7.15 are consistent with the results of the first test: the female group has a disadvantage, and the differences in model scores are significant ($p < 0.01$) based on Wilcoxon rank-sum testing. Effect size for the female group is 0.43, which is moderate-to-strong. Although the results from the previous test provide a convincing explanation for the lower success rates for the female group, they do not necessarily account for all cross-gender variance, as the model still favors inputs with male sponsors when controlling for all other features. The practical effect sizes are small - close to a 1% average difference in scores after the softmax layer for each group, and a small net difference in predicted labels.

Table 7.15: Counterfactual Fairness Results for Resolution Data

Gender	N	Avg Chg.	Score \uparrow	Score \downarrow	To 1	To 0
Male	427	.0120	320	107	5	7
Female	1448	-.0080*	355	1089	19	23

We again observe a preference for a specific group in the congressional dataset. This preference is consistent in all three tests and is at least partially explained by differences in stereotypical features associated with each group. The disadvantage for female sponsors persists even when controlling for the resolution’s subject, cosponsors, and other features.

The congressional dataset is relatively simple, and we cannot expect its features to express the full complexity of the political process. There are almost certainly many additional factors that influence decision-making in Congress that are unaccounted for in the data, and that in turn may affect model fairness. As the results in Table 7.12 indicate, the model predictions closely mirror the base success rates for each group. This dataset is derived from a political process, which may be inherently unfair with respect to gender, in contrast to a decision process governed by an institution. This raises the question of whether a model in this domain should be more fair than the decision process it is trained on. Answering this question is beyond the scope of this work, but the tests we conducted have identified discrepancies for further investigation.

7.3 Conclusion

We found significant outcome differences for subgroups in all of the sensitive variables we examined, which supports our hypothesis. Differences in model predictions across subgroups generally followed the true outcomes. The model for the public trust data had smaller group-level differences than the true outcomes, most likely due to the high degree of feature selection that we performed, which stripped many demographic markers and their proxies.

Counterfactual fairness results were similar to the group-level measures, as the outcome disadvantage associated with one subgroup was present when a different dataset element was assigned to the same subgroup. This is inconsistent with our ideal for individual fairness, in which the outcome should be invariant after subgroup reassignment. However, the differences between groups tended to be small, often less than one percentage point when comparing softmax layer outputs. Domain expertise is needed to evaluate the practical significance of the counterfactual tests and group-level tests, as we have no strong rationale to determine acceptable ranges for cross-group discrepancies. End user stakeholders, which would include policymakers for a decision process, would have to make such a decision.

Having identified subgroups that are disadvantaged in tests 1 and 3, the feature correlations in test 2 offer starting points for future investigation. Using the public trust results as an example, the male subgroup had consistently lower outcome scores. Of the feature types associated with this subgroup, some (e.g., military experience) are of unclear significance to the decision process. Others (drug and

alcohol use) are clearly negative. Not all features will be as clear-cut, and domain experts may be needed to separate features that are less meaningful to the decision process from those that hold decision value. For the features that are useful, an investigation might include a cost/benefit analysis to determine whether the features should continue to be used. Alternatively, the feature may be an explicit part of the decision process (e.g., drug use), and the fact that it occurs more frequently in a subgroup does not constitute unfairness.

Process improvement, which we consider to be a responsibility of the model purveyor stakeholders, is a potential positive outcome of the fairness tests. In the trademark dataset, we found that trademark applications from Californian entities had lower success rates than any of the other locations. The same trademark applications contained more features related to entertainment than the other subgroups. If this outcome discrepancy was deemed to be unfair by domain experts, a further investigation into the process for evaluating entertainment-related trademarks may be warranted. This might result in a change to the evaluation process or a notification to future applicants that the trademark landscape for entertainment companies is crowded, and stronger justifications will be required for registration.

We assert that the builder of a classification model is responsible for answering basic questions about fairness before a solution can be considered for practical use. The three-part test that we proposed offers a repeatable way to answer questions about fairness with empirics. The results alone are probably insufficient to determine if a solution is fair, but they provide a foundation for constructive discussions with stakeholders to determine solution permissibility. The results are

also a starting point to investigate the fairness of the decision process and develop improvements. This may have been previously infeasible for decision tasks that involved substantial amounts of unstructured data, in which traditional approaches such as regression studies would be difficult to apply.

8 LLM Experiments

In this chapter, we detail the use of LLMs to compress the input feature space and generate rationales for the final prediction, with the goal of improving classification accuracy and providing an interpretable summary of the classification decision.

One of the key challenges for classification across all three datasets (public trust clearances, trademark registration, and congressional legislation) is that the feature sets are unbounded in length and can grow to thousands or tens of thousands of tokens. The length and variety of the features disadvantages encoder-type transformer models, as the features may be out of domain and often well in excess of context window length. This is one reason why our recurrent models outperform transformers such as BERT, as reported in Section 5.3.1.2.

We are motivated to explore the application of LLMs to produce a more compact expression of features, with the assumption that the features can be summarized to preserve key factors that drive the classification decision while reducing the length of less pertinent information. We are further interested in using the LLM to incorporate relevant outside knowledge and apply rules or guidelines from the human decision process with the intent of identifying latent features and making them explicit. One of our key assumptions in Chapter 3 was that any solution would need to use imperfect or incomplete input data; outside knowledge may create a more complete feature set and improve accuracy. We will test summarization

and incorporation of outside knowledge in this chapter. In terms of stakeholders, improvements to accuracy would be of interest to end users, whereas successful feature compression or use of outside knowledge offers starting points for model purveyors to refine their technical approaches.

We consider Dutt et al. (2024) as a basis for this line of inquiry, in which an LLM generates a rationale from a dialogue snippet. The explanation augments the dialogue features for use in a downstream classifier for two social meaning detection tasks. In this work, we apply an LLM to generate a combination of summary and commentary on the input features, and then combine the LLM’s outputs with the input features or use them in isolation in a text classification task.

Alternative approaches with rationales include Rajani et al. (2019) and Nye et al. (2021), in which decoder-type transformers were fine-tuned on rationales generated by human annotators or via a rule-based template. These approaches do not fit our use cases, as we do not have the benefit of enlisting annotators through services like Mechanical Turk. Trained examiners execute the real-world decision processes for two of the datasets (public trust clearances and trademarks), and this expertise is not openly available. For the Congressional legislation dataset, the decisions are the result of a political process that involves negotiation and is less bound by rubrics or guidelines; human annotation would most likely require specialized domain expertise to be relevant.

8.1 Model Selection

In applying an LLM to this problem space, we want to maintain consistency with a key constraint expressed in Section 4, in which any solution should be executable on a lower-resourced system. We limit our selection of LLMs to smaller models in the seven to eight billion parameter range. We explored the use of several LLMs that fit this constraint, including Llama3 8B (Dubey et al., 2024), Starling-7B (Zhu et al., 2023), Mistral 7B (Jiang et al., 2023), and Llama 2 7B (Touvron et al., 2023b), using the instruct-tuned versions of these models as appropriate. We selected models that satisfied resource constraints with the highest scores on a public leaderboard¹ based on the work of Chiang et al. (2024). We performed a limited, qualitative evaluation of model responses to 20 randomly selected members from each dataset. Llama 3 8B generated consistently better responses than the others, and was the only model that did not make occasional errors in interpreting prompts and input data, particularly on longer documents.

8.2 Prompting Strategy and Model Execution

We use zero-shot prompting across all datasets. Although few-shot prompting would be preferable, we face two key limitations. The first is that we have no ideal summaries or rationales for use in few-shot prompts. Even if such examples existed, we are further limited by Llama 3’s context window size. Many dataset members require truncation to fit in the window, and the addition of multiple

¹<https://lmarena.ai/>

examples with outputs would further reduce the window size available for the target input. Furthermore, few-shot prompts quickly become long and complex on these datasets; even with notional few-shot examples, the model is prone to generating inconsistent responses and conflating input features with the examples. Fine-tuning may offer some approaches to overcome these limitations, and we discuss some possibilities for future work in the end of this section.

We run Llama 3 in vllm (Kwon et al., 2023), which enforces model-specific chat templates and handles batching. We emphasize the use of prompts that take advantage of the model’s existing domain knowledge. We probe this knowledge by asking open-ended questions about the domain and noting the model’s responses, and then reuse key language from the responses in our prompts at inference time. By referencing the model’s pre-trained knowledge, we avoid enumerating domain-specific rules and guidelines in our prompts, which results in much shorter and simplified instructions and improved response consistency. In the following subsections, we discuss the development of dataset-specific prompts.

8.2.1 Public Trust Clearance Dataset

Our prompting for public trust data draws on information that the US government has published about its adjudication processes for various clearance types. Standards for public trust clearances vary by agency, and not all agencies publish their standards online. However, the guidelines for security clearances, which are

publicly available², form a basis for public trust decisions.

We probe the model for knowledge of the guidelines for security clearances, which are also referred to as the National Security Adjudicative Guidelines and the Security Executive Agent Directive 4 (SEAD-4). An example prompt and response are given in Appendix D.2. The model demonstrates some domain knowledge in this response, but it was only able to correctly name (or provide semantic equivalents for) seven of the 13 major guidelines. The remaining portions of the response were relevant but referred to sub-issues under the main guidelines. When questioned about specific guidelines in the SEAD-4, the model generates reasonable explanations. These results indicate the model has limited domain knowledge and requires more specificity in the prompt to generate desired results.

To compensate for this, we use a prompt that provides more explicit instructions for how to evaluate an input document. We adapt concepts and terminology from the SEAD-4 guidelines to fit the public trust use case. Instead of providing the full feature set to the model, we provide only the document from Source B (the detailed personal history). This document is the most relevant to the federal guidelines, as the other document sources consist of metadata or lack specificity to cleanly map to the guidelines. Source B also comprises the longest documents in the dataset and includes a mix of structured, semi-structured, and unstructured text; this complexity makes it the best candidate for LLM summarization. We provide the prompt template below in Table 8.1, which we tailor to the adjudicat-

²<https://www.dni.gov/files/NCSC/documents/Regulations/SEAD-4-Adjudicative-Guidelines-U.pdf>

ing agency specified in the input data. The *Output Format* given in the table is an example.

As with the trademark dataset, we reformat the raw data in Source B to provide clear descriptions for each feature. This creates a more human-readable structure that is more likely to be in-domain for the LLM than the terse features used by a classifier.

Table 8.1: Prompt Template for Public Trust Data

Role	You are an expert adjudicator for government public trust clearances.
Instructions	The following document contains information from a public trust clearance application to <agency>. Evaluate the document on the basis of <agency>’s adjudicative guidelines.
Output Format	<p>Your response should address each of the following criteria. Limit your response to 1-2 sentences for each criterion.</p> <ul style="list-style-type: none">* Misconduct or negligence in employment* Criminal or dishonest conduct* Material intentional false statement, deception or fraud in employment* Refusal to furnish testimony as required* Alcohol abuse* Illegal drug use* Knowing and willful engagement in acts/activities designed to overthrow the U.S. Government by force.* Statutory or regulatory bar to employment
Document	<input features>

We cannot offer an example response to the prompt due to the proprietary nature of the dataset. However, the model was highly consistent in generating

pertinent responses. The syntax and word choice in its responses closely resembled language used in actual adjudications, which strongly indicated that the Llama 3 training corpus included real case documents.

8.2.2 Trademark Dataset

The trademark dataset contains a large number of features that are essentially metadata, rather than the original documents that an entity would submit in an application to the USPTO. This makes the dataset more difficult to interpret, so rationales generated by the LLM have the potential to significantly improve explainability. We probe the model’s understanding of the trademark process by asking it about the criteria that the USPTO applies when evaluating an application. An example prompt and response are given in Appendix D.1. The model’s responses were consistent across variations in the prompt and temperature settings, which suggests that the model will be reliable in applying this knowledge at inference time.

We draw on the results of these questions to construct a prompt for the trademark dataset, in which we ask the model to comment on the strengths and weaknesses of the application given a set of input features. By prompting the LLM to remark on the application, we intend for the model to apply the USPTO’s standard criteria and potentially surface information is new or otherwise not explicitly encoded in the original features. We also expect the response to summarize key features from the input. The features, as used in the classification task, are

terse and most likely out of domain for an LLM. We reformat them so that a clear description accompanies every feature. We concatenate the features from each document source (case metadata, owner details, control sequence, and usage statement), with each feature on a separate line. The table below shows how we structure the prompts for the model.

Table 8.2: Prompt for Trademark Data

Role	You are an expert trademark examiner in the USPTO with experience handling applications from domestic and foreign entities.
Task Overview	The following document contains information on a trademark application to the USPTO. The document is not a complete application, but instead consists of case metadata, a list of events during the application, and a portion of the statement that describes the applicant’s business use cases.
Instructions	Review the document and provide commentary on the strengths and weaknesses of the application. Base your commentary on the criteria that examiners use to determine trademark registration eligibility.
Output Format	Provide your response in this format, and limit the response length to 300 words: Strengths <your response> Weaknesses <your response> Overall assessment: <your response>
Document	<input features>

8.2.3 Congressional Resolutions

The Congressional resolution dataset is unique in that there are no established guidelines or rubrics for the real-world decision process. Both houses of Congress follow standard procedures for discussions, votes, and committee hearings; however, the substance of these proceedings are do not necessarily adhere to a common rule set. Because of this, we prompt the LLM to apply commonsense reasoning on political topics and request that it offer commentary on a resolution’s likelihood of passing.

We posit that an understanding of historical policy priorities in Congress, along with knowledge of contentious issues of the day, constitute a useful starting point for commonsense political reasoning. We probe the model’s understanding of the these topics and show examples in Appendix D.3. The model demonstrates a general understanding of major topics and the political atmosphere from prior years and Congressional terms. However, it tends to mismatch names and details. In the first prompt, for example, comprehensive immigration reform was a major issue in 2007, but the model cites a bill (with a slightly different name) that was introduced in the Senate; the relevant House bill had a different name. Among other issues, the bill citation numbers are all incorrect, and the Judicial Nomination Reform Act does not appear to exist. The second prompt yields similar results, with the model generating broadly correct statements but making mistakes on some specifics. For example, Lisa Monaco is the deputy attorney general and some of the vote margins are inaccurate. As example outputs will demonstrate,

the model’s general understanding of American politics is sufficient to generate reasonable responses.

We take advantage of the model’s general political knowledge to prompt it for a prediction and commentary on whether or not a Congressional resolution will pass. We prompt the model to apply external, historical knowledge in order to test if such data improves classification accuracy. The model displays a clear bias for predicting that resolutions successfully pass, which is counter to the true outcome distribution. We reduce this bias by adding an additional instruction to the prompt. Although the model only predicts the correct outcome approximately 60 percent of the time, we are more interested in the commentary and its affect on a downstream classifier. The relatively low accuracy is an indicator that the LLM has not memorized the true outcomes in a way that will trivialize the classification task.

We provide two examples of the model’s output for the Congressional resolution dataset in Appendix D.3, illustrating a negative and positive prediction. The model’s reasoning for each prediction appears sound, and the response style is representative of the remainder of the dataset.

8.3 Experimental Design and Results

We now investigate whether the feature summaries/rationales for each dataset improve classification accuracy when added to the base features. We treat LLM-generated text as a separate document source in each dataset and train an ensemble

Table 8.3: Prompt for Congressional Resolution Data

Role	You are an expert on US national politics and Congress.
Task Overview	You will be provided a document that contains information on a Congressional resolution discussed in the House or Senate, along with instructions on how to respond.
Instructions	From the information in the following document, make an educated guess about whether the resolution was successfully passed or failed to pass. Consider which party held the majority in the chamber and the kinds of issues that were a focus at the time. Keep in mind that most resolutions do not pass. Provide a brief explanation for your prediction
Document	<input features>

classifier on the augmented features. We compare the accuracy and f1 scores to those from the original feature sets and report averaged results from five seeded runs.

We also examine classifier accuracy when using the LLM data in isolation as the primary features source. For the trademark and Congressional datasets, we train a classifier on just the LLM-generated responses and the each dataset’s control sequence. For the public trust data, the LLM features are based on Source B, so we use the LLM features in place of Source B in the ensemble.

Table 8.4: Comparison among LLM-augmented Features and Baselines

Dataset	Features	Dev		Test	
		Acc	f1	Acc	f1
Public Trust	Base	.8820	.8787	.8775	.8722
	+LLM	.8792	.8752	.8731	.8660
	LLM only	.8758	.8727	.8716	.8661
Trademark	Base	.9247	.9391	.9236	.9394
	+LLM	.9238	.9397	.9214	.9377
	LLM only	.7430	.7935	.7468	.8003
Resolution	Base	.9466	.9417	.9408	.9344
	+LLM	.9440	.9385	.9296	.9215
	LLM only	.8415	.8254	.8288	.8090

Across all datasets, we observe a slight decrease in accuracy and f1 score when adding the LLM-generated text to the standard set of features. This is an indicator that our attempts to apply commonsense reasoning or add outside information using the LLM did not surface information-bearing features on most datasets. In addition, there is a substantial decrease in accuracy on the trademark and Congressional datasets when using only the LLM features and control sequence for prediction. The LLM’s summarization process is not conditioned on the downstream classifier’s weighting and representation of the input features, so it not

surprising that there is information loss in the summarized output.

One challenge with summarizing or generating rationales for the trademark dataset is that the underlying features comprise many binary metadata fields. During feature selection for the classification task, these fields proved to be useful. However, the degree to which many of the fields contribute to the LLM’s response is unclear. One possible solution is to explicitly prompt the model to handle each metadata field in a specific way, based on further research into the trademark registration process or by probing the classifier’s feature representation. However, it would be cumbersome to enumerate over 40 specific instructions for each metadata field, and the result may be at best an approximation of the classifier’s learned representation.

For the public trust data, the accuracy decrease for the *LLM only* condition is much smaller, as the LLM output replaced one document source. We applied a more detailed prompt, based on published federal guidelines for the actual decision process, to generate the LLM output; this may have resulted in a more efficient use of the raw features and less information loss.

The LLM output on the Congressional resolution dataset is straightforward to compare to the original features. The model’s application of commonsense reasoning (e.g., a congratulatory resolution for a university sports team is politically non-controversial) or introduction of outside knowledge (LGBTQ issues are historically contentious in Congress and unpopular with Republicans) may not be providing new information to the classifier, which is likely to be encoding this information during training. For example, if congratulatory motions for sports

teams frequently pass, we would expect this pattern to be learned. There are also sufficient features to associate specific topics with the majority or minority party, which can be inferred from the *sponsor* and *cosponsor* features, particularly if the association occurs frequently. It is also possible that the LLM’s assertions, while reasonable, may be unsupported in the data, in which case the generated text may be introducing noise.

8.4 Conclusion and Further Investigation

The primary limiting factor in applying an LLM to these datasets is that we have no ideal responses for fine-tuning or few-shot prompting. LLM responses from zero-shot prompting are persuasive for a human reader and offer a degree of explainability. However, they are limited in their capacity to improve classification accuracy when added to the dataset and do not offer comparable accuracy levels when used in place of the full data.

A secondary challenge is that the LLM’s text generation process is isolated from the downstream classification task; ideally, the LLM’s output would reflect the classifier’s learned decision process. One possible solution framework would involve training the classifier and fine-tuning the LLM together, such the gradient from the classification task propagates to the LLM’s weights. Such an arrangement would require reinforcement learning to allow backpropagation through an LLM’s decoding process, but would be beyond the scope of the resource constraints for this work.

Zelikman et al. (2022) develop a prompting-based approach that they show to be "an approximation of an RL-style policy gradient objective." Of the two methods for rationale generation that they introduce, one may be of particular interest to our use cases. They instruct an LLM to generate a "rationalization," in which they provide the correct answer as a hint in the prompt and request the LLM to generate a rationale. The rationalizations are then used to fine-tune the LLM (with the hints removed). This approach offers a potential solution to the lack of data for fine-tuning or few-shot prompting. It also opens a new point of comparison to evaluate the LLM's prediction accuracy against trained classifiers, in addition to augmenting the base feature set with the rationale.

The RGPT method introduced in Zhang et al. (2024) offers an alternative approach via fine-tuning an ensemble of LLMs arranged in a pipeline ("recurrent ensemble") to process input documents and generate a prediction. Although more computationally expensive than our methods, an initial investigation with smaller LLMs would offer a useful starting point. Unlike the previously discussed methods that use rationales, this approach only requires labeled data. The use of our multi-document datasets, which differ greatly from the classification datasets used in the study, should make for a useful indicator of how well this approach generalizes to more complicated classification tasks. CARP, as introduced by (Sun et al., 2023), offers an approach that focuses on kNN search and in-context learning, which lowers the required compute resources. CARP focuses on using the LLM to identify key terms and clues in the document, and it demonstrated strong results on general-domain public datasets. This raises the question of how well the method

extends to specialized domains and more complicated decision processes, which merits investigation on the datasets that we have introduced.

9 Conclusion

9.1 Summary

In this thesis, we introduced a class of decision processes that typically require human expertise to evaluate multiple documentary sources of information. Such tasks are often characterized by the data sources arriving over time. The steps of the decision process, including the arrival of documents, may be journaled in what we term a control sequence, which comprises a sequence of timestamps and events. We formulate the prediction task as a variant of text classification, with a key hypothesis that an ensemble of classifiers will offer better performance, and potentially better explainability, than a single model that takes a concatenation of all input documents. Our formulation of the ensemble assigns a separate classifier to each document type and generates predictions from the fused vector representations of each classifier. We also introduce constraints on the solution space, particularly in terms of computational resources. This is to disallow extremely large models that, while potentially high-performing, would be impractical to deploy in real-world environments where a solution might be fielded.

We developed three datasets to evaluate the ensemble against multiple single model baselines. The first dataset was the primary motivation for the task, and it consists of documents from the US federal government hiring process that are used to evaluate job applicants for public trust clearances. These include two

document types submitted by the applicant and two that are generated during the evaluation process, including a control sequence. The second and third datasets are composed of publicly released government data: one contains data from the USPTO’s trademark registration process, and the other documents resolutions that were introduced for action in the US House or Senate. Both datasets are simpler than the public trust data in terms of feature counts and variety, partly owing to their public nature. This is a net positive, as the three datasets offer diversity in their feature characteristics and complexity.

We demonstrated that the ensemble approach is superior to the single model baselines on the public trust and trademark data, but not on the congressional resolution data. Superficially, we can conclude that the ensemble is preferable for datasets that are more complex. By probing the outputs of the individual classifiers in the ensemble, we can determine the amount of disagreement. The better performing datasets have more disagreement among the classifiers, which we can approximate to how independent the features are across document types in the dataset. Our exploration of the control sequences yielded similar results. The public trust control sequence contained richer information and its event sequence was less likely to be driven by exogenous events. In general, ideal datasets for the ensemble approach contain document types with minimal feature overlap and or collinearity. This can be difficult to measure *a priori*, but a review of the dataset contents, particularly with the assistance of a domain expert, can be of great use to understanding the suitability of a dataset to a given approach.

Because we are applying models to real world decision processes, a model that

is placed into practical use without an understanding of its biases has the potential to inflict harm if it disadvantages a given subgroup. We identify one or more variables of concern for each dataset and apply a three step fairness test to each. We identify areas of concern in each model-dataset pair, although effect sizes for discrimination tended to be low. We also found a high degree of correlation between each variable of concern and other input features. More importantly, each area of concern highlighted aspects of the decision process that warrant further investigation of bias. There remains an open question of whether outcome discrepancies for a given subgroup constitute discrimination, a question that is beyond the scope of this work to answer. We contend that the benefit of such a quantitative analysis is that it provides a clear framing of the issue and its potential magnitude, which may help to motivate future studies.

We also undertook a preliminary examination of using LLMs with each dataset. We focused on rationale generation and the application of common-sense reasoning to augment or simplify feature set. This approach was necessary to work within our established constraints on compute resources. Our initial approaches did not yield results that improved model accuracy, but the generated rationales appear to be useful for explaining the decision outcomes. As detailed in Section 8.4, we identified multiple approaches that are more resource-intensive but should yield interesting results on our datasets and better define how LLMs can be effectively applied to this problem space.

Our findings demonstrated that it is feasible to model each of the decision tasks as text classification problems, and that for a sufficiently complex dataset,

an ensemble of text classifiers outperforms state-of-the-art models while remaining within strict (single-GPU) resource constraints. To conclude this section, we offer a general repeatable process for preparing datasets, training an ensemble, and performing initial fairness evaluations.

Dataset Requirements and Preparation We first need to evaluate the task and its data for feasibility before proceeding. The decision must have a finite number of outcome labels to choose from, and the set of outcomes cannot change over time. Although not tested in this thesis, a real-valued outcome should also be feasible. The following properties and conditions must be satisfied:

- The dataset consists of multiple types of documents, which may mix structured and unstructured data.
- Time series data (control sequences) are acceptable and valued, including data sources that mix temporal data with unstructured text.
- The same document sources must be present in every member of the dataset.
- Data must be cleaned and verified for consistency. For example, text output from optical character recognition systems must be checked for noise and cleaned prior to use.
- For data formats that have changed over time, a canonical schema should be selected and enforced.

- Necessary preprocessing steps should be determined ahead of time, and must be feasible given the time and resources allowed for implementation.

Feature Selection Raw data sources may include information that should not be provided to the model, such as intermediate outcomes that might artificially inflate accuracy. Domain expertise is recommended, as important nuances may not be otherwise apparent.

- Prior to training, determine which features are most important to domain experts at decision time and prioritize their inclusion in the feature set.
- Determine features that might correlate with protected variables or otherwise increase the risk of training an unfair model. Mark these features for exclusion.
- For datasets with control sequences, determine the start and end points in the decision process, during which the model should generate predictions.
- Identify features that might trivialize the task because they are outcome proxies or become available after the critical period outlined above. Remove these features from the dataset.
- Using common principles such as leave one out, determine the marginal value of features in the dataset through iterative training runs using a preliminary modeling approach. Depending on stakeholder needs, one may optimize for accuracy, parsimony/efficiency, or a combination of factors.

Model Selection and Training This process assumes that the ensemble approach is being used, and that multiple classifier modules are considered for each data source.

- Select the classifier modules to consider for use in the ensemble. This thesis used LSTM and CNN-based classifiers. Depending on resource availability, encoder-type transformer models or LLMs may warrant consideration.
- Evaluate classifier module performance on each data source separately, and select the best-performing module.
- Pretrain each classifier module on its associated data source and determine if pretraining improves accuracy. Pretraining typically benefited control sequences in the datasets that we considered.
- Train the full ensemble end-to-end and measure initial performance.
- Experiment with merging data sources to determine if accuracy improves. Alternatively, consider splitting off unique or large features into a separate data source and observing effects on accuracy.
- If time and resources allow, compare the ensemble approach to strong, single-model baselines to ensure that performance is better than simpler approaches.

Performance Evaluation The practitioner will need to determine whether the solution meets the threshold for practical implementation. This is domain- and task-specific, and domain experts should be consulted to understand what the

minimal performance level might be. Accuracy metrics may be of limited use, and expert evaluation can be a helpful complement or replacement.

- In addition to raw accuracy metrics, determine if stakeholders have interest in performance on certain types of cases; provide the appropriate metrics.
- Use accuracy metrics to search for common error cases. Domain experts may have ideas for addressing these; some may be the result of data issues. Error cases should be provided to stakeholders to define limitations of the approach.
- Examine model performance throughout the decision process with an eye toward identifying potential efficiencies. If the model is successful at predicting negative outcomes early in the process for a subset of cases, these cases might be referred for early closure to save resources. Identifying at-risk cases early may enable interventions that promote positive outcomes.

Fairness Evaluation Rather than repeat the description of fairness tests here, we recommend replicating the procedures described Chapter 7.

- In concert with stakeholders and domain experts, identify sensitive variables or groups of concern to examine during testing. Domain experts may have further hypotheses at the subgroup or feature level that require specialized testing.
- Test results, even if non-significant, should inform stakeholder thinking about

the fairness of the decision process and the solution. Hold an open discussion about whether the solution is permissible given the results.

- Identify subgroups or features of concern and facilitate investigations to determine if discrepancies constitute unfairness. Domain expertise is likely to be required.
- Consider if changes to the decision process should be explored to ameliorate fairness concerns. Determine what follow-up tests are needed to enable an investigation, and what metrics should be collected to determine if a remediation is effective.

9.2 Future Work

The experiments in this thesis were tailored to the needs of the stakeholder groups that we outlined in the introduction. As we consider areas to explore in future work, we can relax constraints on resource usage to broaden the space of potential solutions.

Increasing ensemble model diversity. We limited our ensemble solution to two relatively lightweight neural classifier modules: an LSTM with custom regularization and a CNN model with multi-headed self-attention. Encoder transformer models were already available at the start of this work, but including multiple instances of this model type in the ensemble (while supporting long documents) would have been prohibitively expensive. Hypotheses to investigate include:

- An ensemble of large, state-of-the-art transformer classifiers will outperform the current ensemble as well as a single-model approach using the same transformer.
- Adapting these transformers to the task domain through masked language model training (or an equivalent process) will close the performance gap that we observed with BERT when compared to the LSTM and CNN models.

In order to test these, we would select a stable of transformer models to examine, with a preference for models with longer context windows such as longformer (Beltagy et al., 2020) or t5 (Raffel et al., 2023). Pretraining for domain adaptation may be an important step. Due to the semi-supervised nature of this task, we may be able to collect a much larger set of documents than would be possible for the classification task, because we do not need to ensure completeness of dataset elements and outcomes. Our testing would first compare model accuracy for each document source in a given dataset. In keeping with the experimental approach in this work, we would then select the best performing models for the ensemble and repeat our accuracy measurements on the full dataset.

In addition to the encoder-type transformers discussed above, LLMs could also be considered. Zhang et al. (2024), which we also mentioned in the previous chapter, uses an ensemble of LLMs for text classification. We propose adapting this methodology to our datasets by using a separate LLM instance for each data source, fine-tuning each LLM on the data source, and measuring the same accuracy metrics. LLMs in the 70 billion parameter range and above would be considered

for comparisons against encoder-type transformers.

Survival models for control sequences. We relied on a single modeling technique using an LSTM to generate predictions from the control sequence. To add further diversity to an ensemble, we propose the exploration of survival models, which analyze time-to-event data. In our datasets, the event in question is the final outcome, which is always present and therefore avoids complications of left-censored and right-censored data. Recent studies have combined survival analysis with deep learning models to enable the use of multimodal data, including images, unstructured text, and structured data. Some examples include Thorsen-Meyer et al. (2022), which uses a deep learning model to extract features from x-rays, medical records, and clinical measurements, as well as an LSTM to generate the survival analysis. Giunchiglia et al. (2018) also uses recurrent networks with a more limited set of input features, and Hu et al. (2021) introduce a custom transformer architecture that achieves state-of-the-art results. Hypotheses to investigate are:

- A deep survival model’s use of time-to-event data increases the amount of relevant information available when generating predictions and will improve ensemble accuracy over the current approach.
- An alternative architecture, in which an ensemble of classifiers vectorizes the document sources and the deep survival model generates the final predictions, will improve accuracy over a baseline ensemble.

For the first hypothesis, the survival model replaces or complements the LSTM that we currently use for time series data; each of the three survival models cited

above will be incorporated into the ensemble and their vector representation used in generating a final prediction. For the second hypothesis, we propose extending the approach from Thorsen-Meyer et al. (2022), using our existing classifier modules (or other transformers, as discussed previously) to vectorize data sources for input to the survival model.

Our expectation is that time-to-event data provide an additional output to optimize against. The ensemble receives more information during loss calculations, resulting in a better representation of the data. The model’s time-to-event predictions should be useful to end user stakeholders, who will want to understand the time remaining in a decision process before case closure.

Incorporating domain expertise. Access to domain expertise for the public trust dataset informed our approach to feature selection, which most likely provided a more efficient path to a solution. It also contributed to the removal of unnecessary features that may have had adverse effects on model fairness; model predictions showed less of a difference across protected subgroups (relative to the true outcomes) than the predictions for the other datasets. However, we do not know if this difference is the result of feature selection informed by domain expertise or simply due to properties inherent to the datasets and their associated decision tasks.

We propose an experiment to measure the beneficial effects of access to domain expertise for similar decision tasks. Our hypothesis is that access to a domain expert will enable a better understanding of the feature space for a decision task,

leading to a more rapid development of a solution, higher solution accuracy, and lower group-level variance in predicted outcomes in comparison to a control that has no access to an expert. In order to conduct this experiment, we require at least two datasets (preferably three or more) that share the same characteristics as the three examined in this work. The experiment for each dataset would involve the following:

- Two separate development processes, one with a domain expert, the other without. Different individuals or teams implement each solution, with the domain expert randomly assigned to each team.
- Both processes follow a common methodology for feature selection, but they face no restrictions on how to represent features and document sources.
- The number of training runs needed to arrive at a complete feature set and solution will be tracked for comparison, as will the trajectory of improvement for accuracy metrics.
- Accuracy and fairness metrics (using the same fairness tests as in Chapter 7) will also be compared.

With the above experiment, we want to quantify the marginal value of access to a domain expert during solution development. Although the expectation is that access to expertise is a net benefit, the expert may exert an anchoring effect that limits how widely the solution space is searched during development. With experiments on three or more datasets, we hope to identify cases where a domain expert is more (or less) helpful during development.

Bibliography

- A. Adhikari, A. Ram, R. Tang, and J. Lin. Rethinking complex neural network architectures for document classification. 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 4046–4051, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1408. URL <https://aclanthology.org/N19-1408>.
- F. Barbieri, J. Camacho-Collados, F. Ronzano, L. Espinosa-Anke, M. Ballesteros, V. Basile, V. Patti, and H. Saggion. SemEval 2018 task 2: Multilingual emoji prediction. In M. Apidianaki, S. M. Mohammad, J. May, E. Shutova, S. Bethard, and M. Carpuat, editors, *Proceedings of the 12th International Workshop on Semantic Evaluation*, pages 24–33, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/S18-1003. URL <https://aclanthology.org/S18-1003/>.
- B. Becker and R. Kohavi. Adult. UCI Machine Learning Repository, 1996. DOI: <https://doi.org/10.24432/C5XW20>.
- I. Beltagy, M. E. Peters, and A. Cohan. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*, 2020.
- R. Berk. Accuracy and fairness for juvenile justice risk assessments. *Journal of Empirical Legal Studies*, 16(1):175–194, 2019. doi: <https://doi.org/10.1111/jels.12206>. URL <https://onlinelibrary.wiley.com/doi/abs/10.1111/jels.12206>.
- L. Breiman. Bagging predictors. *Machine learning*, 24:123–140, 1996.
- W.-L. Chiang, L. Zheng, Y. Sheng, A. N. Angelopoulos, T. Li, D. Li, H. Zhang, B. Zhu, M. Jordan, J. E. Gonzalez, and I. Stoica. Chatbot arena: An open platform for evaluating llms by human preference, 2024. URL <https://arxiv.org/abs/2403.04132>.
- J. Cohen. *Statistical power analysis for the behavioral sciences*. routledge, 1998.

- R. Collobert and J. Weston. A unified architecture for natural language processing: Deep neural networks with multitask learning. In *Proceedings of the 25th international conference on Machine learning*, pages 160–167, 2008.
- A. Conneau, H. Schwenk, L. Barrault, and Y. Lecun. Very deep convolutional networks for text classification. *arXiv preprint arXiv:1606.01781*, 2016.
- S. Corbett-Davies, E. Pierson, A. Feller, S. Goel, and A. Z. Huq. Algorithmic decision making and the cost of fairness. *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2017. URL <https://api.semanticscholar.org/CorpusID:3228123>.
- X. Dai, I. Chalkidis, S. Darkner, and D. Elliott. Revisiting transformer-based models for long document classification. In Y. Goldberg, Z. Kozareva, and Y. Zhang, editors, *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 7212–7230, Abu Dhabi, United Arab Emirates, Dec. 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.findings-emnlp.534. URL <https://aclanthology.org/2022.findings-emnlp.534/>.
- J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.
- A. Dubey, A. Jauhri, A. Pandey, A. Kadian, A. Al-Dahle, A. Letman, A. Mathur, A. Schelten, A. Yang, A. Fan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.
- R. Dutt, Z. Wu, J. Shi, D. Sheth, P. Gupta, and C. Rose. Leveraging machine-generated rationales to facilitate social meaning detection in conversations. In L.-W. Ku, A. Martins, and V. Srikumar, editors, *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6901–6929, Bangkok, Thailand, Aug. 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.373. URL <https://aclanthology.org/2024.acl-long.373>.
- A. Edwards and J. Camacho-Collados. Language models for text classification: Is in-context learning enough?, 2024. URL <https://arxiv.org/abs/2403.17661>.
- D. R. Gerhardt and J. J. Lee. A tale of four decades: Lessons from uspto trademark prosecution data. *Trademark Rep.*, 112:865, 2022.

- E. Giunchiglia, A. Nemchenko, and M. van der Schaar. Rnn-surv: A deep recurrent model for survival analysis. In *Artificial Neural Networks and Machine Learning–ICANN 2018: 27th International Conference on Artificial Neural Networks, Rhodes, Greece, October 4–7, 2018, Proceedings, Part III 27*, pages 23–32. Springer, 2018.
- S. J. Graham, G. Hancock, A. C. Marco, and A. F. Myers. The uspto trademark case files dataset: Descriptions, lessons, and insights. *Journal of Economics & Management Strategy*, 22(4):669–705, 2013.
- K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.
- S. Hochreiter. Long short-term memory. *Neural Computation MIT-Press*, 1997.
- J. Howard and S. Ruder. Universal language model fine-tuning for text classification. *arXiv preprint arXiv:1801.06146*, 2018.
- S. Hu, E. Fridgeirsson, G. van Wingen, and M. Welling. Transformer-based deep survival analysis. In *Survival Prediction-Algorithms, Challenges and Applications*, pages 132–148. PMLR, 2021.
- H. Ismail Fawaz, G. Forestier, J. Weber, L. Idoumghar, and P.-A. Muller. Deep learning for time series classification: a review. *Data mining and knowledge discovery*, 33(4):917–963, 2019.
- A. Q. Jiang, A. Sablayrolles, A. Mensch, C. Bamford, D. S. Chaplot, D. de las Casas, F. Bressand, G. Lengyel, G. Lample, L. Saulnier, L. R. Lavaud, M.-A. Lachaux, P. Stock, T. L. Scao, T. Lavril, T. Wang, T. Lacroix, and W. E. Sayed. Mistral 7b, 2023. URL <https://arxiv.org/abs/2310.06825>.
- T. Joachims. Text categorization with support vector machines: learning with many relevant features. In *Proceedings of the 10th European Conference on Machine Learning, ECML’98*, page 137–142, Berlin, Heidelberg, 1998a. Springer-Verlag. ISBN 3540644172. doi: 10.1007/BFb0026683. URL <https://doi.org/10.1007/BFb0026683>.
- T. Joachims. Text categorization with support vector machines: Learning with many relevant features. In *European conference on machine learning*, pages 137–142. Springer, 1998b.

- A. E. Johnson, T. J. Pollard, L. Shen, L.-w. H. Lehman, M. Feng, M. Ghassemi, B. Moody, P. Szolovits, L. Anthony Celi, and R. G. Mark. Mimic-iii, a freely accessible critical care database. *Scientific data*, 3(1):1–9, 2016.
- J. Kleinberg, S. Mullainathan, and M. Raghavan. Inherent trade-offs in the fair determination of risk scores, 2016. URL <https://arxiv.org/abs/1609.05807>.
- S. Krishna, R. Gupta, A. Verma, J. Dhamala, Y. Pruksachatkun, and K.-W. Chang. Measuring fairness of text classifiers via prediction sensitivity, 2022. URL <https://arxiv.org/abs/2203.08670>.
- A. Krogh and J. Vedelsby. Neural network ensembles, cross validation, and active learning. *Advances in neural information processing systems*, 7, 1994.
- L. I. Kuncheva and C. J. Whitaker. Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy. *Machine learning*, 51:181–207, 2003.
- W. Kwon, Z. Li, S. Zhuang, Y. Sheng, L. Zheng, C. H. Yu, J. E. Gonzalez, H. Zhang, and I. Stoica. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*, 2023.
- K. Lang. Newsweeder: learning to filter netnews. In *Proceedings of the Twelfth International Conference on International Conference on Machine Learning*, ICML’95, page 331–339, San Francisco, CA, USA, 1995. Morgan Kaufmann Publishers Inc. ISBN 1558603778.
- D. D. Lewis, Y. Yang, T. G. Rose, and F. Li. Rcv1: A new benchmark collection for text categorization research. *J. Mach. Learn. Res.*, 5:361–397, Dec. 2004. ISSN 1532-4435.
- Z. Lipton, J. McAuley, and A. Chouldechova. Does mitigating ml’s impact disparity require treatment disparity? In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. URL https://proceedings.neurips.cc/paper_files/paper/2018/file/8e0384779e58ce2af40eb365b318cc32-Paper.pdf.
- Z. C. Lipton, D. C. Kale, C. Elkan, and R. Wetzel. Learning to diagnose with lstm recurrent neural networks. *arXiv preprint arXiv:1511.03677*, 2015.

- 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, 2019. URL <https://arxiv.org/abs/1907.11692>.
- V. Lynn, N. Balasubramanian, and H. A. Schwartz. Hierarchical modeling for user personality prediction: The role of message-level attention. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5306–5316, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.472. URL <https://aclanthology.org/2020.acl-main.472>.
- A. L. Maas, R. E. Daly, P. T. Pham, D. Huang, A. Y. Ng, and C. Potts. Learning word vectors for sentiment analysis. In D. Lin, Y. Matsumoto, and R. Mihalcea, editors, *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 142–150, Portland, Oregon, USA, June 2011. Association for Computational Linguistics. URL <https://aclanthology.org/P11-1015/>.
- M. E. Maron. Automatic indexing: An experimental inquiry. *J. ACM*, 8(3): 404–417, July 1961. ISSN 0004-5411. doi: 10.1145/321075.321084. URL <https://doi.org/10.1145/321075.321084>.
- S. Merity, N. S. Keskar, and R. Socher. Regularizing and optimizing lstm language models. *arXiv preprint arXiv:1708.02182*, 2017.
- F. Mosteller and D. L. Wallace. Inference in an authorship problem: A comparative study of discrimination methods applied to the authorship of the disputed federalist papers. *Journal of the American Statistical Association*, 58(302):275–309, 1963.
- J. J. Nay. Predicting and understanding law-making with word vectors and an ensemble model. *PLOS ONE*, 12(5):e0176999, May 2017. ISSN 1932-6203. doi: 10.1371/journal.pone.0176999. URL <http://dx.doi.org/10.1371/journal.pone.0176999>.
- K. Nigam, A. McCallum, S. Thrun, and T. Mitchell. Learning to classify text from labeled and unlabeled documents. In *Proceedings of the Fifteenth National/Tenth Conference on Artificial Intelligence/Innovative Applications of Artificial Intelligence*, AAAI '98/IAAI '98, page 792–799, USA, 1998. American Association for Artificial Intelligence. ISBN 0262510987.

- M. Nye, A. J. Andreassen, G. Gur-Ari, H. Michalewski, J. Austin, D. Bieber, D. Dohan, A. Lewkowycz, M. Bosma, D. Luan, C. Sutton, and A. Odena. Show your work: Scratchpads for intermediate computation with language models, 2021. URL <https://arxiv.org/abs/2112.00114>.
- J. Ooms. *pdftools: Text Extraction, Rendering and Converting of PDF Documents*, 2023. <https://docs.ropensci.org/pdftools/> (website) <https://github.com/ropensci/pdftools> <https://poppler.freedesktop.org> (upstream).
- H. Park, Y. Vyas, and K. Shah. Efficient classification of long documents using transformers. In S. Muresan, P. Nakov, and A. Villavicencio, editors, *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 702–709, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-short.79. URL <https://aclanthology.org/2022.acl-short.79/>.
- F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.
- D. Premasiri, T. Ranasinghe, and R. Mitkov. Can model fusing help transformers in long document classification? an empirical study. In R. Mitkov and G. Angelova, editors, *Proceedings of the 14th International Conference on Recent Advances in Natural Language Processing*, pages 871–878, Varna, Bulgaria, Sept. 2023. INCOMA Ltd., Shoumen, Bulgaria. URL <https://aclanthology.org/2023.ranlp-1.94/>.
- H. Pugaliya, J. Route, K. Ma, Y. Geng, and E. Nyberg. Bend but don’t break? multi-challenge stress test for qa models. In *Proceedings of the 2nd Workshop on Machine Reading for Question Answering*, pages 125–136, 2019.
- S. Purpura and D. Hillard. Automated classification of congressional legislation. In *Proceedings of the 2006 International Conference on Digital Government Research*, dg.o ’06, page 219–225. Digital Government Society of North America, 2006. doi: 10.1145/1146598.1146660. URL <https://doi.org/10.1145/1146598.1146660>.

- C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer, 2023. URL <https://arxiv.org/abs/1910.10683>.
- N. F. Rajani, B. McCann, C. Xiong, and R. Socher. Explain yourself! leveraging language models for commonsense reasoning. In A. Korhonen, D. Traum, and L. Màrquez, editors, *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4932–4942, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1487. URL <https://aclanthology.org/P19-1487>.
- R. E. Schapire. The strength of weak learnability. *Machine learning*, 5:197–227, 1990.
- W. M. Schuster, M. Marcowitz-Bitton, and D. R. Gerhard. An empirical study of gender and race in trademark prosecution. *Southern California Law Review*, 94:1407–1466, 2021. URL https://scholarship.law.unc.edu/cgi/viewcontent.cgi?article=1564&context=faculty_publications.
- H. Schütze, D. A. Hull, and J. O. Pedersen. A comparison of classifiers and document representations for the routing problem. In *Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '95, page 229–237, New York, NY, USA, 1995. Association for Computing Machinery. ISBN 0897917146. doi: 10.1145/215206.215365. URL <https://doi.org/10.1145/215206.215365>.
- K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.
- R. Socher, A. Perelygin, J. Wu, J. Chuang, C. D. Manning, A. Ng, and C. Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In D. Yarowsky, T. Baldwin, A. Korhonen, K. Livescu, and S. Bethard, editors, *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1631–1642, Seattle, Washington, USA, Oct. 2013. Association for Computational Linguistics. URL <https://aclanthology.org/D13-1170/>.
- X. Sun, X. Li, J. Li, F. Wu, S. Guo, T. Zhang, and G. Wang. Text classification via large language models, 2023. URL <https://arxiv.org/abs/2305.08377>.

- M. Suzgun, L. Melas-Kyriazi, S. K. Sarkar, S. D. Kominers, and S. M. Shieber. The harvard uspto patent dataset: A large-scale, well-structured, and multi-purpose corpus of patent applications, 2022. URL <https://arxiv.org/abs/2207.04043>.
- H.-C. Thorsen-Meyer, D. Placido, B. S. Kaas-Hansen, A. P. Nielsen, T. Lange, A. B. Nielsen, P. Toft, J. Schierbeck, T. Strøm, P. J. Chmura, et al. Discrete-time survival analysis in the critically ill: a deep learning approach using heterogeneous data. *NPJ digital medicine*, 5(1):142, 2022.
- H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, N. Bashlykov, S. Batra, P. Bhargava, S. Bhosale, D. Bikel, L. Blecher, C. C. Ferrer, M. Chen, G. Cucurull, D. Esiobu, J. Fernandes, J. Fu, W. Fu, B. Fuller, C. Gao, V. Goswami, N. Goyal, A. Hartshorn, S. Hosseini, R. Hou, H. Inan, M. Kardas, V. Kerkez, M. Khabsa, I. Kloumann, A. Korenev, P. S. Koura, M.-A. Lachaux, T. Lavril, J. Lee, D. Liskovich, Y. Lu, Y. Mao, X. Martinet, T. Miyahirov, P. Mishra, I. Molybog, Y. Nie, A. Poulton, J. Reizenstein, R. Rungta, K. Saladi, A. Schelten, R. Silva, E. M. Smith, R. Subramanian, X. E. Tan, B. Tang, R. Taylor, A. Williams, J. X. Kuan, P. Xu, Z. Yan, I. Zarov, Y. Zhang, A. Fan, M. Kambadur, S. Narang, A. Rodriguez, R. Stojnic, S. Edunov, and T. Scialom. Llama 2: Open foundation and fine-tuned chat models, 2023a. URL <https://arxiv.org/abs/2307.09288>.
- H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, N. Bashlykov, S. Batra, P. Bhargava, S. Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023b.
- Y. Xiao and K. Cho. Efficient character-level document classification by combining convolution and recurrent layers. *CoRR*, abs/1602.00367, 2016. URL <http://arxiv.org/abs/1602.00367>.
- P. Yang, X. Sun, W. Li, S. Ma, W. Wu, and H. Wang. Sgm: sequence generation model for multi-label classification. *arXiv preprint arXiv:1806.04822*, 2018.
- Z. Yang, D. Yang, C. Dyer, X. He, A. Smola, and E. Hovy. Hierarchical attention networks for document classification. In *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: human language technologies*, pages 1480–1489, 2016.

- Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. R. Salakhutdinov, and Q. V. Le. Xlnet: Generalized autoregressive pretraining for language understanding. *Advances in neural information processing systems*, 32, 2019.
- A. W. Yu, D. Dohan, M. Luong, R. Zhao, K. Chen, M. Norouzi, and Q. V. Le. Qanet: Combining local convolution with global self-attention for reading comprehension. *CoRR*, abs/1804.09541, 2018. URL <http://arxiv.org/abs/1804.09541>.
- E. Zelikman, Y. Wu, J. Mu, and N. Goodman. Star: Bootstrapping reasoning with reasoning. *Advances in Neural Information Processing Systems*, 35:15476–15488, 2022.
- X. Zhang, J. Zhao, and Y. LeCun. Character-level convolutional networks for text classification, 2016. URL <https://arxiv.org/abs/1509.01626>.
- Y. Zhang, M. Wang, C. Ren, Q. Li, P. Tiwari, B. Wang, and J. Qin. Pushing the limit of llm capacity for text classification, 2024. URL <https://arxiv.org/abs/2402.07470>.
- B. Zhu, E. Frick, T. Wu, H. Zhu, and J. Jiao. Starling-7b: Improving llm helpfulness & harmlessness with rlaiif, November 2023.

APPENDICES

A Example Dataset Contents

A.1 Public Trust Dataset

Algorithm 1 Pseudocode for Constructing Public Trust Dataset Instance

```

initialize  $source_a$ ,  $source_b$ ,  $source_c$ ,  $source_d$ ,  $source_e$ 
 $outcome \leftarrow \text{Null}$ 
while  $outcome$  is Null do
     $new\_event, timestamp, payload \leftarrow poll\_for\_event()$ 
    if  $payload$  is of type  $A$  then
         $source_a \leftarrow payload$ 
    else if  $payload$  is of type  $B$  then
         $source_b \leftarrow payload$ 
    else if  $payload$  is of type  $C$  then
         $source_c \leftarrow payload$ 
    else if  $payload$  is of type  $D$  then
        append  $payload$  to  $source_d$ 
    end if
    append ( $new\_event, timestamp$ ) to  $source_e$ 
    if  $new\_event$  is an ending condition then
        if  $new\_event$  is "candidate hired" then
             $outcome \leftarrow 1$ 
        else
             $outcome \leftarrow 0$ 
        end if
    end if
end while

```

A.2 Trademark Dataset

```
'acq_dist_in': '0',
'acq_dist_part_in': '0',
'amend_lb_for_app_in': '0',
'amend_lb_for_reg_in': '0',
'amend_lb_itu_in': '0',
'amend_lb_use_in': '0',
'amend_principal_in': '0',
'amend_reg_dt': '',
'amend_supp_reg_in': '0',
'cancel_pend_in': '0',
'cert_mark_in': '0',
'cfh_status_cd': '700',
'cfh_status_dt': '2022-11-08',
'chg_reg_in': '0',
'coll_memb_mark_in': '0',
'coll_serv_mark_in': '0',
'coll_trade_mark_in': '0',
'concur_use_in': '0',
'concur_use_pend_in': '0',
'concur_use_pub_in': '0',
'draw_3d_cur_in': '0',
'draw_3d_file_in': '0',
'draw_color_cur_in': '0',
'draw_color_file_in': '0',
'exm_attorney_name': 'EXAMPLE A. ATTORNEY',
'exm_office_cd': '020',
'file_location': 'PUBLICATION AND ISSUE SECTION',
'file_location_dt': '2022-11-08',
'filing_dt': '2021-09-23',
```

Figure A.1: Example *case_file* Excerpt (Names Anonymized)

```
[{'event_cd': 'ABN6',
  'event_dt': '2001-11-28',
  'event_seq': '8',
  'event_type_cd': 'S',
  'serial_no': '75763656'},
 {'event_cd': 'NOAM',
  'event_dt': '2000-10-31',
  'event_seq': '7',
  'event_type_cd': 'O',
  'serial_no': '75763656'},
 {'event_cd': 'PUB0',
  'event_dt': '2000-08-08',
  'event_seq': '6',
  'event_type_cd': 'A',
  'serial_no': '75763656'},
 {'event_cd': 'NPUB',
  'event_dt': '2000-07-07',
  'event_seq': '5',
  'event_type_cd': 'O',
  'serial_no': '75763656'},
 {'event_cd': 'CNSA',
  'event_dt': '2000-05-03',
  'event_seq': '4',
  'event_type_cd': 'P',
  'serial_no': '75763656'},
 {'event_cd': 'CRFA',
  'event_dt': '2000-03-20',
  'event_seq': '3',
  'event_type_cd': 'I',
  'serial_no': '75763656'},
 {'event_cd': 'CNRT',
  'event_dt': '2000-01-05',
  'event_seq': '2',
  'event_type_cd': 'F',
  'serial_no': '75763656'},
 {'event_cd': 'DOCK',
  'event_dt': '1999-12-27',
  'event_seq': '1',
  'event_type_cd': 'D',
  'serial_no': '75763656'}]
```

Figure A.2: Short Example of *event* Data

```
[{'own_addr_1': '',
  'own_addr_2': 'Horseferry House, Horseferry Road',
  'own_addr_city': 'London',
  'own_addr_country_cd': 'GB',
  'own_addr_other_cd': '',
  'own_addr_postal': 'SW1P2AW',
  'own_addr_state_cd': '',
  'own_altn_name': '',
  'own_composed_of': '',
  'own_entity_cd': '3',
  'own_entity_desc': '',
  'own_id': '22707',
  'own_nalty_country_cd': 'GB',
  'own_nalty_other_cd': '',
  'own_nalty_state_cd': '',
  'own_name': 'BURBERRY LIMITED',
  'own_seq': '1',
  'own_type_cd': '40',
  'serial_no': '71281718'},
 {'own_addr_1': '',
  'own_addr_2': '18/22 HAYMARKET',
  'own_addr_city': 'LONDON',
  'own_addr_country_cd': '',
  'own_addr_other_cd': 'GB2',
  'own_addr_postal': 'SW1Y 4DQ',
  'own_addr_state_cd': '',
  'own_altn_name': '',
  'own_composed_of': '',
  'own_entity_cd': '3',
  'own_entity_desc': '',
  'own_id': '22708',
  'own_nalty_country_cd': 'GB',
  'own_nalty_other_cd': '',
  'own_nalty_state_cd': '',
  'own_name': 'BURBERRYS LIMITED',
  'own_seq': '1',
  'own_type_cd': '30',
  'serial_no': '71281718'}]
```

Figure A.3: Example *owner* Data

```
[{'serial_no': '71235488',
  'statement_text': 'The following goods "steam turbines; steam condensers" is '
                    'divided to child application #75/983732.',
  'statement_type_cd': 'A00001'},
{'serial_no': '71235488',
  'statement_text': '[ STEAM ACCUMULATORS, STEAM ENGINES, STEAMHAMMERS, '
                    'MACHINE PLANTS FOR RIVER STEAMERS AND SEAGOING VESSELS, ] '
                    '[ STEAM TURBINES, ] [ WATER TURBINES, ] [ STEAM '
                    'CONDENSERS; ][ PISTON COMPRESSORS FOR COMPRESSING STEAM, '
                    'AIR, OR OTHER LIQUIDS; ROTARY COMPRESSORS FOR COMPRESSING '
                    'STEAM, AIR, OR OTHER LIQUIDS; ROAD ROLLERS, LIFTING AND '
                    'TRANSPORTING MACHINERY, EXCAVATORS, HYDRAULIC PRESSES; '
                    'AND HYDRAULIC PLANTS, PRINTING PRESSES, BEET DRYERS '
                    'UTILIZING THE SMOKE GASES OF THE BOILERS, PLANTS FOR FAST '
                    'DRYING OF SUGAR LOAVES AND CUBES WITH HOT-AIR '
                    'CIRCULATION, DRYING MACHINES IN THE TEXTILE INDUSTRY, '
                    'DRYING APPARATUS WITH DRUMS FOR INDUSTRIAL DISTILLERIES, '
                    'DRYING PLANTS FOR COAL AND DRYING APPARATUS IN '
                    'PULVERIZED-COAL EQUIPMENT, DRYING PLANTS FOR HOPS; CREAM '
                    'SEPARATORS; APPARATUS AND PLANTS NECESSARY FOR THE '
                    'MANUFACTURE AND WORKING OF GREASE-NAMELY, BONE CRUSHERS, '
                    'GREASE COLLECTORS, COOLERS, GREASE-REFINING APPARATUS; '
                    'PLANTS FOR LIXIVIATING SOLUTIONS OF GLUE, VACUUM '
                    'APPARATUS FOR GLUE, REFINING APPARATUS FOR GLUE, '
                    'COOLINGTABLES FOR SLABS OF GLUE AND DRYING CHAMBERS FOR '
                    'THE SAME; COPPER AUTOCLAVES, IRON AUTOCLAVES WITH COPPER '
                    'LINERS, AND ROTATING AUTOCLAVES FOR FAT SPLITTING; '
                    'SEPARATORS AND WASHING MACHINES FOR FATTY ACIDS, REFINING '
                    'AND MIXING APPARATUS, MIXING APPARATUS FOR FATTY ACIDS, '
                    'VACUUM DISTILLATION APPARATUS FOR STEARIN, MELTING AND '
                    'MIXING APPARATUS FOR PURE STEARIN, HYDRAULIC PRESSES FOR '
                    'COLD AND HOT STEARIN, EVAPORATING PLANTS FOR GLYCERINE '
                    'WITH PRECIPITATION AND RECOVERY OF SODIUM CHLORIDE, '
                    'DISTILLING APPARATUS FOR PURE GLYCERINE, EXTRACTION '
                    'APPARATUS FOR STEARIN; COOLING MACHINES, CUTTING '
                    'MACHINES, DRYING CHAMBERS, MACHINES FOR PRODUCING SOAP '
                    'SHAVINGS, MIXING APPARATUS AND STAMPING MACHINES FOR '
                    'TOILET SOAP, PRESSES FOR FAT RESIDUES, MACHINES FOR '
                    'MAKING MARGARINE AND BUTTER, MACHINES FOR PRODUCING '
                    'LUBRICANTS; APPARATUS FOR MANUFACTURING AND WORKING '
                    'VEGETABLE AND MINERAL OILS-NAMELY, CLEANING AND SORTING '
                    'DEVICES FOR SEEDS, MACHINES FOR REMOVING HUSKS AND THE '
                    'LIKE, ROLLIN...',
  'statement_type_cd': 'GS0231'}]
```

Figure A.4: Example *statement* Data


```
[{'for_appl_country_cd': 'CA',
  'for_appl_filing_dt': '',
  'for_appl_no': '',
  'for_appl_other_cd': '',
  'for_appl_priority_in': '0',
  'for_appl_seq': '1',
  'for_registration_dt': '1934-09-27',
  'for_registration_exp_dt': '1993-09-27',
  'for_registration_no': 'N.S.13/4,224',
  'for_renewal_dt': '',
  'for_renewal_exp_dt': '',
  'for_renewal_no': '',
  'serial_no': '73226507'},
 {'for_appl_country_cd': 'CA',
  'for_appl_filing_dt': '',
  'for_appl_no': '',
  'for_appl_other_cd': '',
  'for_appl_priority_in': '0',
  'for_appl_seq': '2',
  'for_registration_dt': '1974-07-12',
  'for_registration_exp_dt': '1989-07-12',
  'for_registration_no': '200,473',
  'for_renewal_dt': '',
  'for_renewal_exp_dt': '',
  'for_renewal_no': '',
  'serial_no': '73226507'},
 {'for_appl_country_cd': 'CA',
  'for_appl_filing_dt': '',
  'for_appl_no': '',
  'for_appl_other_cd': '',
  'for_appl_priority_in': '0',
  'for_appl_seq': '3',
  'for_registration_dt': '1975-01-17',
  'for_registration_exp_dt': '1990-01-17',
  'for_registration_no': '204,700',
  'for_renewal_dt': '',
  'for_renewal_exp_dt': '',
  'for_renewal_no': '',
  'serial_no': '73226507'}]
```

Figure A.5: Example *foreign-app* Data

A.3 Congressional Resolution Dataset

```

<billStatus>
  <version>3.0.0</version>
  <bill>
    <number>230</number>
    <updateDate>2023-01-11T13:24:02Z</updateDate>
    <updateDateIncludingText>2023-01-11T13:24:02Z</updateDateIncludingText>
    <originChamber>Senate</originChamber>
    <originChamberCode>S</originChamberCode>
    <type>SRES</type>
    <introducedDate>2013-09-17</introducedDate>
    <congress>113</congress>
    <committees>
      ...
    </committees>
    <actions>
      <item>
        <actionDate>2013-09-17</actionDate>
        <committees>
          ...
        </committees>
        <sourceSystem>
          ...
        </sourceSystem>
        <text>Referred to the Committee on Rules and Administration.</text>
        <type>IntroReferral</type>
      </item>
      <item>
        <actionDate>2013-09-17</actionDate>
        <committees>
          ...
        </committees>
        <sourceSystem>
          ...
        </sourceSystem>
        <text>Committee on Agriculture, Nutrition, and Forestry. Original measure reported to Senate by Senator Stabenow. Without written report. (text of measure as reported in Senate: CR S6528)</text>
        <type>Committee</type>
      </item>
      <item>
        <actionDate>2013-09-17</actionDate>
        <text>Committee on Agriculture, Nutrition, and Forestry. Original measure reported to Senate by Senator Stabenow. Without written report. (text of measure as reported in Senate: CR S6528)</text>
        <type>Committee</type>
        <actionCode>14000</actionCode>
        <sourceSystem>
          ...
        </sourceSystem>
        <committees>
          ...
        </committees>
      </item>
      <item>
        <actionDate>2013-09-17</actionDate>
        <text>Introduced in Senate</text>
        <type>IntroReferral</type>
        <actionCode>10000</actionCode>
        <sourceSystem>
          ...
        </sourceSystem>
      </item>
    </actions>
  </bill>
</billStatus>

```

Figure A.6: Example XML Data for *Basic Details* and *Actions*

```

▼<sponsors>
  ▼<item>
    <bioguideId>S000770</bioguideId>
    <fullName>Sen. Stabenow, Debbie [D-MI]</fullName>
    <firstName>Debbie</firstName>
    <lastName>Stabenow</lastName>
    <party>D</party>
    <state>MI</state>
    <isByRequest>N</isByRequest>
  </item>
</sponsors>
▶<policyArea>
  ...
</policyArea>
▼<subjects>
  ▼<legislativeSubjects>
    ▼<item>
      <name>Congressional committees</name>
    </item>
    ▼<item>
      <name>Senate Committee on Agriculture, Nutrition, and Forestry</name>
    </item>
  </legislativeSubjects>
  ▼<policyArea>
    <name>Congress</name>
  </policyArea>
</subjects>
▼<summaries>
  ▼<summary>
    <versionCode>00</versionCode>
    <actionDate>2013-09-17</actionDate>
    <actionDesc>Introduced in Senate</actionDesc>
    <updateDate>2013-09-20T20:47:09Z</updateDate>
    ▼<text>
      <![CDATA[ <p>Authorizes expenditures by the Senate Committee on Agriculture, Nutrition, and Forestry for the 113th Congress. </p> ]]>
    </text>
  </summary>
  ▼<summary>
    <versionCode>80</versionCode>
    <actionDate>2013-09-17</actionDate>
    <actionDesc>Reported to Senate without amendment</actionDesc>
    <updateDate>2013-09-20T21:04:31Z</updateDate>
    ▼<text>
      <![CDATA[ <p><b>(This measure has not been amended since it was introduced. The summary of that version is repeated here.)</b></p> <p>Authorizes expenditures by the Senate Committee on Agriculture, Nutrition, and Forestry for the 113th Congress. </p> ]]>
    </text>
  </summary>
</summaries>
<title>An original resolution authorizing expenditures by the Committee on Agriculture, Nutrition, and Forestry.</title>
▼<titles>
  ▼<item>
    <titleType>Display Title</titleType>
    <title>An original resolution authorizing expenditures by the Committee on Agriculture, Nutrition, and Forestry.
  </title>
  </item>
  ▼<item>
    <titleType>Official Title as Introduced</titleType>
    <title>An original resolution authorizing expenditures by the Committee on Agriculture, Nutrition, and Forestry.
  </title>

```

Figure A.7: Example XML Data for *Sponsors*, *Subjects*, *Summaries*, and *Titles*

B Supplementary Experimental Results

Table B.1: Confusion Matrix for Ensemble (Left) and Single Model LSTM (Right) for Public Trust Self-Cancel Outcomes

	Ensemble		Single LSTM	
	Label 1	Label 0	Label 1	Label 0
Pred. 1	1	54	2	62
Pred. 0	1	176	0	168

Table B.2: Confusion Matrix Comparison for Public Trust Applicants who Cancel Post-Clearance

	Ensemble		Single LSTM	
	Label 1	Label 0	Label 1	Label 0
Pred. 1	1	32	2	35
Pred. 0	1	32	0	29

Table B.3: Confusion Matrix Comparison for Public Trust Applicants with Inconsistent Source A/B Data

	Ensemble		Single LSTM	
	Label 1	Label 0	Label 1	Label 0
Pred. 1	202	59	191	65
Pred. 0	16	330	27	324

Table B.4: Confusion Matrix Comparison for Public Trust DQ Type 2 Cases

	Ensemble		Single LSTM	
	Label 1	Label 0	Label 1	Label 0
Pred. 1	1	62	1	68
Pred. 0	0	400	0	394

Table B.5: Confusion Matrix for Ensemble with Auxiliary Outputs on Trademark Data

	Label 1	Label 0
Pred. 1	3446	351
Pred. 0	185	2017

Table B.6: Confusion Matrix for Ensemble when Trademark Test Set Predictions Disagree with Single Model

	Label 1	Label 0
Pred. 1	1	124
Pred. 0	28	76

Table B.7: Confusion Matrix for Ensemble with Auxiliary Outputs on Resolution Data

	Label 1	Label 0
Pred. 1	769	65
Pred. 0	78	963

Table B.8: Confusion Matrix for Ensemble (Left) and Single Model LSTM (Right) on Majority Party Resolutions

	Ensemble		Single LSTM	
	Label 1	Label 0	Label 1	Label 0
Pred. 1	424	31	435	39
Pred. 0	39	469	28	461

Table B.9: Confusion Matrix for Ensemble (Left) and Single Model LSTM (Right) on Minority Party Resolutions

	Ensemble		Single LSTM	
	Label 1	Label 0	Label 1	Label 0
Pred. 1	344	15	362	21
Pred. 0	40	513	22	507

Table B.10: Agreement between Classifier Modules and Full Ensemble

Classifier	ρ	Jaccard
Public Trust		
Source A	.8553	.7455
Source B	.5087	.4579
Source D	.4021	.4874
Source E	.5800	.5826
Trademark		
Case File	.5963	.6601
Owner	.6753	.7285
Statement	.5980	.7112
Events	.6500	.7360
Resolution		
Basic Data	.6496	.5853
Sponsor	.7491	.6623
Summary	.6951	.6383
Actions	.8985	.7758

C Model Parameters

For the regularized LSTM classifier module, we list all parameters following the terminology used in Merity et al. (2017). The same parameters were used across datasets.

Table C.1: Regularized LSTM Parameters

Parameter	Value
Embedding size	256
Embedding dropout	0.2
Weight-drop	0.1
Hidden size	512
LSTM layers	2
Temporal activation weighting (β)	0.1
MLP layer dropout	0.5
Batch size	64
Learning rate	0.001
EMA decay factor	0.99
Optimizer	Adam

Parameters for the CNN with self-attention classifier module are given below. All parameters not listed (e.g., CNN kernel sizes) follow the baseline design given

in Yu et al. (2018).

Table C.2: CNN Classifier Parameters

Parameter	Value
Word embedding size	300
Character embedding size	200
Hidden size	512
Attention heads	8
Dropout	0.1
Batch size	64
Learning rate	0.001
EMA decay factor	0.99
L2 weight decay	5×10^{-8}
Learning rate warmup steps	1000
Optimizer	Adam

D LLM Prompts and Responses

D.1 Trademark Dataset

What criteria do examiners use at the US Patent and Trademark Office when evaluating a trademark application?

The USPTO (United States Patent and Trademark Office) uses the following criteria to evaluate a trademark application:

1. ****Distinctiveness****: Is the mark distinctive and not generic or descriptive? In other words, is the mark unique and not simply a description of the goods or services?
2. ****Likelihood of Confusion****: Will the mark be confused with existing trademarks or marks that are likely to be registered in the future? The examiner will consider the similarity of the marks, the similarity of the goods or services, and the degree of consumer recognition of the marks.
3. ****Descriptiveness****: Is the mark descriptive of the goods or services? If so, is it a common term or phrase that is used in the industry?
4. ****Functionality****: Does the mark serve a functional purpose, such as the shape of a product or the color of a product?
5. ****Geographical Significance****: Is the mark a geographical name, such as a city or state name? If so, is it a well-known geographical name?
6. ****Immoral or Scandalous****: Is the mark immoral, scandalous, or disparaging?
7. ****Likelihood of Dilution****: Will the mark dilute the reputation of another mark, even if the marks are not similar?

If the examiner finds that the mark does not meet these criteria, the application will be refused. If the examiner finds that the mark meets these criteria, the application will be approved and the mark will be registered.

Example set of input features and response using prompt template from Table 8.2.

Input features:

Collective membership mark indicator: no
 Collective service mark indicator: no
 Collective trademark indicator: no
 Color drawing currently: no
 Color drawing at filing: no
 3D drawing currently: no
 3D drawing at filing: no
 Service mark indicator: yes
 Standard characters claim: no
 Amendment to foreign application priority basis: no
 Amendment to foreign registration basis: no
 Amendment to intent to use basis: no
 Amendment to use basis: no
 Cancellation proceeding pending: no
 Certification mark indicator: no
 Any amendment, correction, less goods, less classes: no
 Subject to concurrent use: no
 Concurrent use proceeding pending: no
 Use application at filing: yes
 Foreign registration basis: no
 International registration basis currently: no
 Foreign application priority basis currently: no
 Foreign registration basis at filing: no
 International registration basis at filing: no
 No basis claim currently: no
 Contains foreign application info and priority claim: no
 Intent to use application at filing: no
 Actual characters of mark text: AAA
 Opposition proceeding pending: no
 Amend to principal register: no
 Published subject to concurrent use: no
 Affidavit of incontestability acknowledged: yes
 Acquired distinctiveness entry: no
 Acquired distinctiveness entry to part of mark: no

Amend to supplemental register: no
Filed for or registered on supplemental register: no
Trademark indicator: no
Use application currently: yes
US issued notice of first refusal to IB: no
Words 'and others; appear in list of prior registrations: no
No records for foreign trademark applications
Mark physical characteristics: 1 0 0 0
Owner: 3 AMERICAN AUTOMOBILE ASSOCIATION, INC.
Events during trademark processing:
1987-03-11: REGISTERED - SEC. 9 FILED/CHECK RECORD FOR SEC. 8
1987-05-23: REGISTERED AND RENEWED (FIRST RENEWAL - 20 YRS)
2006-08-30: TEAS SECTION 8 & 9 RECEIVED
2006-08-30: REGISTERED - COMBINED SECTION 8 (10-YR) & SEC. 9 FILED
2006-10-27: ASSIGNED TO PARALEGAL
2006-11-21: REGISTERED - SEC. 8 (10-YR) ACCEPTED/SEC. 9 GRANTED
2006-11-21: REGISTERED AND RENEWED (SECOND RENEWAL - 10 YRS)
2008-02-03: NOTICE OF SUIT
2008-02-03: NOTICE OF SUIT
2008-02-12: NOTICE OF SUIT
2008-05-29: NOTICE OF SUIT
2008-05-30: NOTICE OF SUIT
2008-06-06: NOTICE OF SUIT
2008-06-16: NOTICE OF SUIT
2008-06-18: NOTICE OF SUIT
2008-06-19: NOTICE OF SUIT
2008-06-20: NOTICE OF SUIT
2008-08-04: NOTICE OF SUIT
2008-08-21: NOTICE OF SUIT
2008-08-22: NOTICE OF SUIT
2008-09-03: NOTICE OF SUIT
2008-09-12: NOTICE OF SUIT
2008-10-03: NOTICE OF SUIT
2008-10-28: NOTICE OF SUIT
2009-02-02: NOTICE OF SUIT
2009-02-05: NOTICE OF SUIT
2009-02-06: CASE FILE IN TICRS
2009-02-25: NOTICE OF SUIT

[126 'NOTICE OF SUIT' lines removed]

2014-02-04: NOTICE OF SUIT

Statement of goods/services related to mark: TEACHING MOTOR VEHICLE OPERATION; CONDUCTING TRAFFIC AND PEDESTRIAN SAFETY CAMPAIGNS, AND GIVING TRAFFIC SAFETY LESSONS DISSEMINATING TRAVEL INFORMATION AND MAKING TRAVEL ARRANGEMENTS PROVIDING EMERGENCY ROAD SERVICE PROVIDING BAIL BOND ARRANGING FOR DISCOUNT PURCHASES, COLLECTING DAMAGE CLAIMS; OFFERING REWARDS FOR INFORMATION LEADING TO ARREST AND CONVICTION OF PERSONS STEALING MEMBERS' AUTOMOBILES; AND PLACING INSURANCE WITH UNDERWRITERS AUTOMOBILE ASSOCIATION SERVICES RENDERED TO MOTOR VEHICLE OWNERS, MOTORISTS, AND TRAVELERS GENERALLY-NAMELY, OBTAINING MOTOR VEHICLE LICENSE PLATES AND TITLE CERTIFICATES; SPONSORING SCHOOL SAFETY PATROLS; ADVOCATING LEGISLATION FAVORABLE TO SAFE AND ECONOMICAL MOTOR VEHICLE TRAVEL, OPERATION, AND MAINTENANCE; CONDUCTING MOTOR VEHICLE TESTS AND MAKING TESTS OF AUTOMOTIVE AND RELATED PRODUCTS; AND RATING TOURIST ACCOMMODATIONS

LLM Response:

Strengths:

1. The mark is distinctive and memorable, making it more likely to be recognized by the public.
2. The applicant has a strong reputation and is well-established in the industry, which can increase the likelihood of the mark being registered.
3. The mark is used in commerce and has been used for over 30 years, which can demonstrate its validity and enforceability.

Weaknesses:

1. The mark is a service mark, which may be more difficult to register than a trademark, as it is used to identify the source of a service rather than a product.
2. The mark is used in a variety of industries, including education, insurance, and travel, which may make it difficult to determine the scope of the mark's use and the goods or services it is used to identify.
3. The applicant has filed multiple notices of opposition, which may indicate that there are potential conflicts with other marks or that the applicant is trying to assert its rights over a broader scope of goods or services than initially claimed.

Overall assessment:

Overall, the strengths of this application include its distinctiveness, the applicant's reputation, and the mark's long history of use. However, the weaknesses include the potential difficulties in registering a service mark, the complexity of the mark's use across multiple industries, and the potential conflicts with other marks. As an examiner, I would carefully review the application to determine whether the mark is eligible for registration and whether any potential conflicts or issues with the mark's use should be addressed.

D.2 Public Trust Dataset

Within the context of US government security clearances, list the 13 adjudicative guidelines in the SEAD-4.

The 13 adjudicative guidelines are:

1. Allegiance: The individual's loyalty to the United States.
2. Foreign Influence: The extent to which the individual has been influenced by foreign powers or their agents.
3. Foreign Preference: The extent to which the individual has demonstrated a preference for a foreign power or its interests over those of the United States.
4. Dual Citizenship: The individual's status as a dual citizen of the United States and another country.
5. Foreign Travel: The extent to which the individual has traveled to or resided in a foreign country, particularly if it is a country of concern.
6. Foreign Financial Ties: The extent to which the individual has financial ties to a foreign power or its agents.
7. Conflicts of Interest: The extent to which the individual has conflicts of interest that could compromise their loyalty to the United States.
8. Criminal Conduct: The individual's criminal history, including any criminal conduct that could compromise their loyalty to the United States.
9. Financial Problems: The individual's financial problems, including any debt or financial obligations that could compromise their loyalty to the United States.
10. Emotional Instability: The individual's emotional stability, including any mental health issues that could compromise their ability to perform their duties.
11. Sexual Behavior: The individual's sexual behavior, including any behavior that could compromise their loyalty to the United States.
12. Personal Behavior: The individual's personal behavior, including any behavior that could compromise their loyalty to the United States.
13. Security Clearance History: The individual's history of security clearances, including any previous denials or revocations of clearance.

D.3 Congressional Resolution Dataset

What were the legislative priorities for the Republicans in the US House in 2007? Limit your response to 200 words.

In 2007, the legislative priorities for the Republicans in the US House of Representatives were:

Tax Relief: Republicans sought to extend the 2001 and 2003 tax cuts, which were set to expire in 2010.

Immigration Reform: Republicans pushed for comprehensive immigration reform, including border security and a pathway to citizenship for undocumented immigrants.

Energy Independence: Republicans advocated for increased domestic energy production, including drilling in the Arctic National Wildlife Refuge (ANWR) and the Outer Continental Shelf (OCS).

Judicial Nominations: Republicans sought to confirm conservative judicial nominees to the federal bench.

Regulatory Reform: Republicans aimed to reduce the regulatory burden on businesses and individuals, particularly in areas such as environmental and financial regulation.

These priorities were reflected in several key bills, including: H.R. 2, the "Fair Tax Act of 2007" (tax reform)

H.R. 164, the "Immigration Reform Act of 2007" (immigration reform)

H.R. 322, the "Energy Policy Act of 2007" (energy independence)

H.R. 142, the "Judicial Nominations Reform Act of 2007" (judicial nominations)

H.R. 123, the "Regulatory Reform Act of 2007" (regulatory reform)

What were the most contentious issues in the Senate during the 117th US Congress?

The 117th United States Congress, which convened on January 3, 2021, and adjourned on January 3, 2023, was marked by significant partisanship and contentious issues. Here are some of the most notable ones:

1. **Confirmation of Supreme Court Justice Ketanji Brown Jackson:** The Senate confirmed Justice Jackson to the Supreme Court on April 7, 2022, but not before a contentious hearing and a 53-47 vote, with three Republicans voting in favor of her confirmation.

2. **Biden Administration Nominees:** The Senate confirmed several Biden administration nominees, including:

* Merrick Garland's successor, Lisa Monaco, as Attorney General, with a 98-2 vote.

* Marty Walsh as Secretary of Labor, with a 68-29 vote.

- * Xavier Becerra as Secretary of Health and Human Services, with a 50-49 vote.
 - * Deb Haaland as Secretary of the Interior, with a 51-40 vote.
 - * Gina Raimondo as Secretary of Commerce, with a 92-5 vote.
3. COVID-19 Relief and Stimulus Packages: The Senate debated and passed several COVID-19 relief and stimulus packages, including:
- * American Rescue Plan Act (March 2021): A \$1.9 trillion stimulus package that passed with a 50-49 vote, with Vice President Kamala Harris casting a tie-breaking vote.
 - * Infrastructure Investment and Jobs Act (November 2021): A \$1.2 trillion infrastructure package that passed with a 69-30 vote.
4. Gun Control Legislation: The Senate debated several gun control bills, including:
- * Background Check Expansion Act: A bill that would have expanded background checks for gun purchases, which failed to pass with a 47-50 vote.
 - * Red Flag Law: A bill that would have allowed courts to temporarily confiscate firearms from individuals deemed a risk to themselves or others, which failed to pass with a 47-50 vote.
5. Immigration Reform: The Senate debated several immigration reform bills, including:
- * Dream and Promise Act: A bill that would have provided a pathway to citizenship for undocumented immigrants brought to the United States as children, which failed to pass with a 52-48 vote.
 - * Border Security, Economic Opportunity, and Return Act: A bill that would have provided funding for border security and a pathway to citizenship for undocumented immigrants, which failed to pass with a 55-45 vote.
6. Climate Change and Energy Policy: The Senate debated several climate change and energy policy bills, including:
- * Green New Deal: A non-binding resolution that outlined a plan to transition the United States to a clean energy economy, which failed to pass with a 47-50 vote.
 - * Clean Energy for America Act: A bill that would have provided tax credits for clean energy production, which failed to pass with a 52-42 vote.
7. Voting Rights and Election Reform: The Senate debated several voting rights and election reform bills, including:
- * For the People Act: A comprehensive election reform bill that would have expanded voting rights and campaign finance reform, which failed to pass with a 50-49 vote.
 - * John Lewis Voting Rights Advancement Act: A bill that would have restored the Voting Rights Act of 1965, which failed to pass with a 49-46 vote.
- These are just a few examples of the contentious issues that arose during the 117th US Congress. The Senate's partisanship and gridlock were evident throughout the session,

with many bills failing to pass or being blocked by filibusters.

Example sets of input features and responses using prompt from Table 8.3.

Input features:

Date introduced: 2014-01-13

Chamber: US Senate

Majority Party: Democratic

Legislative Categories: Athletes, Congressional tributes, Higher education, North Dakota, School athletics, Sports and Recreation

Summary of resolution: (This measure has not been amended since it was introduced. The summary of that version is repeated here.) Congratulates the North Dakota State University football team as the champion of for [sic] winning the 2013 National Collegiate Athletic Association Division I Football Championship Subdivision title.

Titles: A resolution congratulating the North Dakota State university football team for winning the 2013 National Collegiate Athletic Association Division I Football Championship Subdivision title.

Sponsor: Sen. Hoeven, John [R-ND]

Cosponsors: Sen. Heitkamp, Heidi [D-ND]

LLM Response:

Based on the information provided, I predict that the resolution was successfully passed. The resolution was introduced in the Senate, where the Democratic Party held the majority. The resolution is a congratulatory measure, which is a relatively non-controversial issue. Additionally, the resolution is focused on a specific university and its athletic team, which may have garnered support from both parties. The fact that the resolution has not been amended since its introduction suggests that it may have been a straightforward and uncontroversial measure.

Input features:

Date introduced: 2017-05-17

Chamber: US House

Majority Party: Republican

Legislative Categories: Commemorative events and holidays, Elementary and secondary education, Foreign aid and international relief, Health care quality, Higher education, Human rights, School administration, Sex, gender, sexual orientation discrimination, Civil Rights and Liberties, Minority Issues

Summary of resolution: Expresses support for the goals and ideals of an International

Day Against Homophobia and Transphobia. Encourages: (1) health care providers to offer culturally and clinically competent care to the lesbian, gay, bisexual, or transgender (LGBT) community; (2) schools to support the creation of gay-straight alliances to achieve safer learning environments; (3) individuals to learn about national resources for those who identify as LGBT, and (4) the federal multilateral aid efforts to prioritize the health and human rights of LGBT people.

Titles: Supporting the goals and ideals of the International Day Against Homophobia and Transphobia.

Sponsor: Rep. Lee, Barbara [D-CA-13] Cosponsors: Rep. Grijalva, Raul M. [D-AZ-3] Rep. Kildee, Daniel T. [D-MI-5] Rep. Roybal-Allard, Lucille [D-CA-40] Rep. Pocan, Mark [D-WI-2] Rep. Pallone, Frank, Jr. [D-NJ-6] Rep. McCollum, Betty [D-MN-4] Rep. Hastings, Alcee L. [D-FL-20] Rep. Cicilline, David N. [D-RI-1] Rep. DelBene, Suzan K. [D-WA-1] Rep. Wasserman Schultz, Debbie [D-FL-23] Rep. Speier, Jackie [D-CA-14] Rep. Norton, Eleanor Holmes [D-DC-At Large] Rep. Smith, Adam [D-WA-9] Rep. Gallego, Ruben [D-AZ-7] Rep. Clark, Katherine M. [D-MA-5] Rep. Brownley, Julia [D-CA-26] Rep. Blumenauer, Earl [D-OR-3] Rep. Watson Coleman, Bonnie [D-NJ-12] Rep. Slaughter, Louise McIntosh [D-NY-25] Rep. McEachin, A. Donald [D-VA-4] Rep. Lowenthal, Alan S. [D-CA-47] Rep. Maloney, Sean Patrick [D-NY-18] Rep. Johnson, Henry C. "Hank," Jr. [D-GA-4] Rep. Deutch, Theodore E. [D-FL-22] Rep. Gutierrez, Luis V. [D-IL-4] Rep. Kilmer, Derek [D-WA-6] Rep. Schakowsky, Janice D. [D-IL-9] Rep. Lowey, Nita M. [D-NY-17] Rep. Pingree, Chellie [D-ME-1] Rep. Sánchez, Linda T. [D-CA-38] Rep. Ellison, Keith [D-MN-5] Rep. Keating, William R. [D-MA-9] Rep. Khanna, Ro [D-CA-17] Rep. Frankel, Lois [D-FL-21] Rep. Quigley, Mike [D-IL-5] Rep. Swalwell, Eric [D-CA-15] Rep. Hanabusa, Colleen [D-HI-1] Rep. Napolitano, Grace F. [D-CA-32] Rep. Larsen, Rick [D-WA-2] Rep. Himes, James A. [D-CT-4] Rep. Yarmuth, John A. [D-KY-3] Rep. Levin, Sander M. [D-MI-9] Rep. Meeks, Gregory W. [D-NY-5] Rep. McGovern, James P. [D-MA-2] Rep. Sires, Albio [D-NJ-8] Rep. Engel, Eliot L. [D-NY-16] Rep. Demings, Val Butler [D-FL-10] Rep. Polis, Jared [D-CO-2] Rep. Panetta, Jimmy [D-CA-20] Rep. Raskin, Jamie [D-MD-8] Rep. Jayapal, Pramila [D-WA-7] Rep. Espaillat, Adriano [D-NY-13] Rep. Davis, Susan A. [D-CA-53] Rep. Perlmutter, Ed [D-CO-7] Rep. Clarke, Yvette D. [D-NY-9] Rep. Huffman, Jared [D-CA-2] Rep. Lynch, Stephen F. [D-MA-8] Rep. Moulton, Seth [D-MA-6] Rep. Correa, J. Luis [D-CA-46] Rep. Kihuen, Ruben J. [D-NV-4] Rep. Vargas, Juan [D-CA-51] Rep. Titus, Dina [D-NV-1] Rep. Lieu, Ted [D-CA-33] Rep. Jackson Lee, Sheila [D-TX-18] Rep. Dingell, Debbie [D-MI-12] Rep. Brady, Robert A. [D-PA-1] Rep. Rush, Bobby L. [D-IL-1] Rep. Shea-Porter, Carol [D-NH-1] Rep. Maloney, Carolyn B. [D-NY-12] Rep. Gomez, Jimmy [D-CA-34] Rep. Esty, Elizabeth H. [D-CT-5] Rep. Higgins, Brian [D-NY-26] Rep. Carson, Andre [D-IN-7] Rep. Soto,

Darren [D-FL-9]

LLM Response:

Based on the information provided, I predict that the resolution did not pass. The resolution was introduced in the House of Representatives, which was controlled by the Republican Party at the time. The resolution focuses on issues related to LGBTQ+ rights, which have historically been a contentious issue in Congress. While there may have been some Democratic support for the resolution, it is unlikely that the Republican majority would have supported it. Additionally, the resolution's focus on issues such as healthcare, education, and human rights may have been seen as too progressive or controversial for some lawmakers. Given these factors, it is likely that the resolution did not pass.