Toward Data-Efficient Multimodal Learning

Liangke Gui

July 2025

School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

Thesis Committee:
Alexander G. Hauptmann, Carnegie Mellon University
Yonatan Bisk, Carnegie Mellon University
Emma Strubell, Carnegie Mellon University
Daniel Fried, Carnegie Mellon University
Po-Yao (Bernie) Huang, Meta AI

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy.

Abstract

Multimodal learning, which integrates information from vision, language, and sound, plays a central role in human perception and cognition. Humans naturally combine inputs from different modalities to understand complex environments, learn from limited examples, and generalize across tasks. Inspired by this ability, recent advances in multimodal learning have led to significant progress in tasks such as visual question answering, image-text retrieval, and multimodal information extraction. Despite these achievements, existing models face key challenges that limit their scalability and applicability in real-world scenarios.

One major limitation is the heavy reliance on large-scale, manually annotated datasets for both pre-training and downstream tasks. Collecting such data is labor intensive, costly, and difficult to scale, especially for complex modalities such as open-ended reasoning and video understanding that require contextual and temporal reasoning. Moreover, these models often struggle to generalize in low-resource settings, where annotated data is limited. In addition, many state-of-the-art models are trained in a closed-book fashion, where all knowledge is stored in the model parameters. This hinders their ability to incorporate external knowledge sources dynamically, such as structured databases or large language models, limiting their flexibility and explainability in open-domain reasoning.

This thesis addresses these limitations by advancing data-efficient multimodal learning through three key strategies. First, we investigate how structured human priors can be embedded into model design and training to improve learning efficiency and generalization in low-data regimes. Second, we explore the use of weak supervision signals, such as natural-occurring image-text pairs and external knowledge bases, to enhance representation learning without relying on extensive manual annotation. Third, we introduce preference-leraning frameworks that leverage large language models to guide training in complex tasks, particularly in video understanding and open-domain reasoning, where traditional labels are difficult to define or scale. Across these components, this thesis aims to reduce the need for explicit supervision while improving model performance, interpretability, and adaptability, contributing to the development of more scalable and robust multimodal AI systems.

Acknowledgments

I would like to express my heartfelt gratitude to my advisor, Professor Alexander Hauptmann, for his invaluable guidance, patience, and encouragement throughout my Ph.D. journey. His deep insights, broad vision, and ability to connect ideas have deeply shaped my research and inspired me to think beyond boundaries. I am especially grateful for his trust and support during challenging times, which continually pushed me to grow and improve. His kindness in sharing knowledge, opportunities, and mentorship has made a lasting impact on both my professional development and personal growth.

I am also grateful to my committee members Yonatan Bisk, Emma Strubell, Daniel Fried, and Po-Yao (Bernie) Huang for their thougtful advice, invaluable feedback, and flexibility throughout the entire process. I am fortunate to have had the opportunity to work with Yonatan, whose insights, ideas, and practical guidance have been critical in shaping my research. His ability to see the bigger picture has continually inspired me to think more broadly. I deeply appreciate Emma and Daniel for their constructive feedback during my dissertation work, which greatly strengthened this dissertation. I am also thankful to Bernie for encouraging me to think deeply and explore further since the very beginning of my Ph.D. journey.

I am also very thankful to my friends and collaborators, both inside and outside CMU, for supporting me in many ways during these years. Your encouragement, company, and understanding gave me strength, whether it was through talking with me, sharing happy moments, or just being there when I faced difficulties. You made my Ph.D. journey easier and more meaningful.

Lastly, but most importantly, I want to thank my family. To my mom, dad, sister, and brother-in-law, thank you for your unconditional love, for teaching me the value of hard work and perseverance, and for always supporting my decisions. I also thank my two cats, Yuan Gui and Pi Gui, although they have no idea what a Ph.D. is, their quiet company beside my desk, playful moments, and warm greetings brought me comfort and strength throughout this journey.

To all of you, my deepest gratitude for shaping not only this dissertation, but also the person I have become.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation of Research

Human perception naturally combines different types of sensory inputs, such as vision, language, and sound, to understand and interact with the world. This ability allows people to quickly learn new concepts, generalize in different situations, and adapt to new tasks with minimal supervision. Inspired by this, the goal of multimodal machine learning is to develop models that can process and reason across multiple modalities in a similar way. Recent advances in large-scale pretraining have led to rapid progress in multimodal tasks like visual question answering(VQA) [9, 80, 166, 206], image-text retrieval [151, 185], image captioning, optical character recognition (OCR) [178, 211], and multimodal information extraction [95, 96]. However, several challenges still limit the scalability and practical use of multimodal systems.

High Dependence on Extensive Human Supervision. Most existing multimodal models [48, 67, 115, 133, 134, 143, 212, 276] rely on large-scale human-annotated datasets for both pretraining and fine-tuning. Although this enables strong performance, creating such datasets is costly and time-consuming, especially for complex tasks like video understanding or open-ended reasoning. Reducing this dependence is the key to making multimodal systems more scalable.

Limited Generalization in Low-Data Scenarios. Despite achieving impressive results in benchmark datasets, multimodal models [49, 67, 115, 133, 143, 276] often struggle when applied to unseen data distributions and novel tasks. Unlike humans, who can use prior knowledge and context to adapt, current models have limited ability to generalize in a few-shot and zero-shot learning settings. Improving this requires the use of human-inspired priors and weak forms of supervision.

Lack of Dynamic Knowledge Integration in Open-Domain Tasks. Most state-of-the-art multimodal models [5, 45, 245, 269] operate in a “closed-book" fashion, where all knowledge is embedded within the model parameters. This limits their flexibility and their ability to incorporate new information or provide explainable reasoning. Open-domain tasks, such as visual question answering and multimodal reasoning, would benefit from models that can access and use external knowledge, including structured databases and large language models.

Challenges in Video Understanding and Temporal Reasoning. Video-based tasks, such as instruction following, captioning, and question answering, introduce additional complexities due to the need for temporal reasoning and contextual understanding. Traditional supervised learning approaches require extensive human annotations, making large-scale video dataset annotation impractical. A promising alternative is to align model predictions with human preferences using large language models, reducing the need for manual labels.

This thesis focuses on improving **data-efficient multimodal learning**, with the goal of reducing human supervision, improving generalization, and enabling models to reason in open-domain and video-based tasks. To achieve this, we explore three key strategies: (1) **incorporating human priors** to improve model adaptability in low-data scenarios; (2) **Leveraging weak supervision** from naturally occurring image-text pairs and external knowledge; and (3) **integrating preference learning** with large language models to guide training, especially in tasks that are difficult to annotate, such as video understanding and open-ended reasoning. These strategies aim to reduce the reliance on labeled data while building more general and practical multimodal systems.

1.2 Thesis Organization

This thesis explores multimodal learning in the data by gradually reducing the dependence on human supervision. It follows a structured approach, beginning with leveraging human priors, then incorporating weakly supervised signals, and ultimately utilizing the implicit knowledge embedded within large language models. Each part builds upon the previous one, forming a coherent progression from manual encoded knowledge to automated learning from large-scale models.

Part I: Human Priors for Data Efficiency The first part of the thesis investigates how human prior knowledge can enhance multimodal learning. Humans naturally use contextual and structural information to recognize patterns and generalize from limited examples. By embedding these priors into machine learning models, we can improve data efficiency and learning effectiveness.

Chapter 2 examines the challenges of handwritten text recognition, particularly Arabic script, which exhibits complex structures and contextual dependencies. This chapter explores how human-inspired priors, such as character dependencies within local contexts, can improve recognition accuracy. By embedding these priors into the learning process, the proposed approach enhances model robustness while requiring fewer labeled examples.

Chapter 3 addresses the learning of a few shots, a setting in which models must generalize from a limited number of labeled samples. This chapter introduces a data augmentation method guided by human priors, leveraging the intuition that similar objects exhibit similar behaviors. By generating realistic variations, the proposed approach improves model performance in low-data scenarios, demonstrating the effectiveness of human-inspired augmentation strategies.

Although human priors provide valuable information to reduce supervision in specific tasks, a more scalable approach is needed for generalizable multimodal learning. The next part of this thesis investigates how weakly supervised signals can be leveraged to improve multimodal representation learning, reducing reliance on explicit annotations.

Part II: Weak Supervision for Multimodal Representation Learning The second part of this thesis explores how weakly supervised signals can serve as an alternative to large-scale labeled data. Instead of relying on explicit human annotations, models can utilize naturally occurring associations, such as image-text pairs, and external knowledge sources to learn meaningful representations with minimal supervision.

Chapter 4 investigates the use of image-text pairs as a source of weak supervision, demonstrating how leveraging loosely aligned data improves multimodal representation learning.

Chapter 5 extends this approach by integrating structured external knowledge bases and implicit knowledge stored in large language models to improve reasoning capabilities, particularly for open-ended tasks such as visual question answering.

Although weak supervision improves multimodal learning across static image-based tasks, video understanding presents additional challenges due to temporal dependencies and the need for sequential reasoning. The final part of this thesis explores how large language models can further reduce annotation costs while improving model adaptability in video-based tasks.

Part III: Preference Learning for Video Understanding The third part of the thesis focuses on using large language models to facilitate data-efficient learning in video-based tasks. Unlike static image-text learning, video understanding requires capturing sequential dependencies and aligning actions with human intent. By incorporating preference learning with large language models, we can mitigate the need for extensive human annotations while improving the adaptability of the model.

Chapter 6 introduces a preference learning framework that leverages large language models to align predictions with human expectations in video-based tasks such as instruction following, captioning, and question answering. Learning from implicit signals rather than explicit labels significantly reduces annotation costs while improving model performance.

Summary of Contributions This thesis contributes to the field of multimodal learning by reducing the reliance on labeled data through the use of human priors, weak supervision, and preference learning. By transitioning from explicit supervision toward learning from broader, less-curated sources, it presents a step-by-step approach for developing more efficient and adaptable models. The findings support both theoretical progress and practical applications, paving the way for more scalable and generalizable AI systems in real-word scenarios.

Part I

Human Priors for Data Efficiency

Chapter 2

Arabic Handwriting Recognition with Human Prior Knowledge

This chapter explores the incorporation of human prior knowledge into the recognition of Arabic handwriting. Specifically, the inherent dependence of a letter on its local context is utilized to improve recognition accuracy. By modeling such contextual dependencies, this work demonstrates how leveraging domain-specific priors can enhance the performance of handwriting recognition systems while reducing reliance on extensive labeled data.

2.1 Overview

Handwritten text recognition has been a ubiquitous research problem in the field of computer vision. Most existing approaches focus on the recognition of handwritten words without considering the cursive nature and significant differences in the writing of individuals. In this paper, we address these problems by leveraging an adaptive context-aware reinforced agent which learns the actions to determine the scales of context regions during inference. We formulate our approach in a reinforcement learning framework. Specifically, we construct the action set with a number of context lengths. Given an image feature sequence, our model is trained to adaptively choose the optimal context length according to the observed state. An attention mechanism is then used to selectively attend the context region. Our model can generalize well from recognizing isolated words to recognizing individual lines of text while remain low computation overheads. We conduct extensive experiments on three large-scale handwritten text recognition datasets. The experimental results show that our proposed model is superior to the state-of-the-art alternatives.

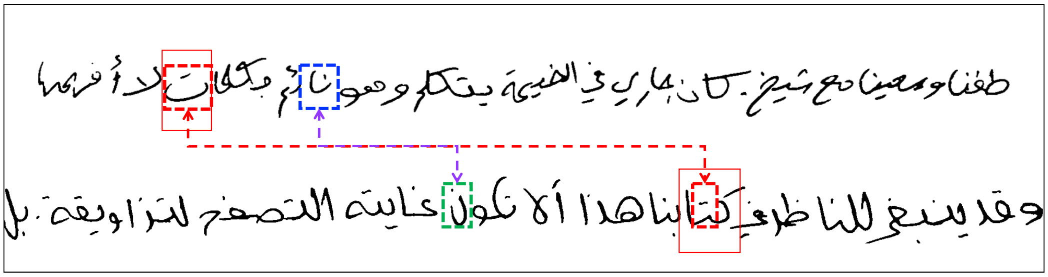


Figure 2.1: Two samples from KHATT dataset. The shape of the same character ت (red) varies under different surrounding context, while two different words of نا (blue) and ن (green) share a similar shape. Correctly inferring a character depends on its correlated characters which we denote as **local context**. We refer to the number of adjacent characters needed to make an inference as **context length.**

2.2 Motivation

Handwritten text recognition (HWR) is commonly used to extract natural languages from images. It remains an open research problem, in which noisy, real-valued input streams are annotated with strings of discrete labels, such as letters or words. Handwritten text recognition presents relevant applications such as bank check reading, mail sorting, and content preservation of historical documents. Due to the importance of these applications, it has attracted increasing research attention in recent years.

Despite recent advances in scene text recognition [29, 52, 132, 209, 236], recognizing handwritten text, due to the cursive nature of handwritten characters and significant differences in the writing of individuals, remains challenging. Several attempts using convolutional neural networks (CNNs) [23, 121, 186] have been shown to produce impressively low error rates on handwritten word datasets. However, these systems use fixed-size CNNs and focus on isolated words which are rarely readily available in real world applications. Another general approach is to use recurrent neural networks (RNNs) associated with connectionist temporal classification (CTC) [82]. They are capable of recognizing a line of text without word-level segmentation. Doetsch *et al* [64] use a stacked bidirectional long short-term memory (BLSTM) [83, 93] with PCA-based features. In a recent German handwritten text recognition competition [201], the top methods use architectures which generally consist of CNNs and RNNs and achieve remarkable performance. Bluche *et al* [26] propose a MDLSTM-attention system to recognize handwritten text from paragraphs by incorporating multi-dimensional LSTM [81] and attention mechanism. We are inspired by this idea but propose significant modifications.

One observation is that the reading order of characters is typically established by convention (*e.g.*, a primary order from right to left in Arabic scripts). Therefore, while LSTM is capable of capturing long-term dependencies in the handwritten text recognition task, the local context around a target position is informative to determine a character, as illustrated in Figure 6.1. Characters may rely on different scales of context region. For example, due to the cursive writing, inferring the character in dash-line box may rely on the context in solid-line box. Meanwhile, within the context region, the characters may contribute differently to the inference. Motivated by this, we propose to introduce an adaptive context length selection and soft attention mechanism into the handwritten text recognition task.

To address the above mentioned issues, we present a framework that treats context regions localization as a decision making process, by which an agent would adaptively select a context length according to the observed states. In our framework, we prepare a number of context lengths as the action set. Choosing the context length is formulated as a reinforcement learning framework. By applying a policy network, an agent learns to select the optimal length of context region by analyzing the observed content. To keep the policy execution lightweight, we take all the decisions in a single step which can be seen as an instantiation of associative reinforcement learning [222]. Thus we maximize the negative loss as the global reward of our policy network.

We refer the proposed framework as Adaptive Context-aware Reinforced Agent. Our contributions are summarized as follows:

- We make the first attempt to address the handwritten text recognition problem in a reinforcement learning framework. By learning an adaptive context-aware reinforced agent, our proposed model is capable of selectively attending context regions during inference.
- Unlike previous work on Arabic words recognition, we solve a more challenging task of Arabic handwritten text line recognition.
- We show that our proposed model generalizes well from isolated words to text lines recognition and achieves the state-of-the-art performances on several benchmarks.

Our paper is structured as follows. We first overview the recent research on handwritten text recognition, attention mechanism, and reinforcement learning in Section 2.3. We then present our model in Section 2.4, followed by a description of experiments in Section 2.5 and results in Section 2.5. We conclude and present future directions in Section 5.7.

2.3 Prior Work

We first discuss widely used approaches for handwritten text recognition. We then discuss the recent advances in attention mechanisms and reinforcement learning which our work builds on.

Handwritten Text Recognition. Traditional approaches to handwritten text recognition are mainly focused on two key elements: the strategy to extract features and the way to decode the output of the classifiers to predict the sequence of characters [215]. Poznanski *et al* [186] propose a CNN-N-Gram model to estimate the n-gram frequency profile given a handwritten word image. Despite of the remarkable performance on several handwritten benchmarks, the manually defined N-gram CNN model has a large number of output nodes which increases the training complexity. Shi *et al* [209] propose a CRNN model to recognize text in the wild and is closely related to our work. In their work, a CNN model is used to extract feature sequences from input images and a recurrent network is built for making prediction for each frame of the feature sequence. While their approach is designed for scene word recognition with a constrained image scale, our model is focused on handwritten text recognition and can generalize from single word to text lines.

Attention Model. “Attention-based" methods have shown to be successful for machine translation [14], image caption generation [53, 261] and speech recognition [32, 54]. Attention-based mechanisms can allow the model to learn alignments between different modalities. Many researchers have explored different attention methods to solve the image-based text recognition task. Deng *et al* [61] propose a coarse-to-fine attention mechanism to convert images into presentational markup by constructing a sparse coarse attention to reduce the number of fine attention cells. To recognize the text in the wild, Lee *et al* [123] propose a R2AM model to selectively exploit image features in a coordinated way by incorporating soft attention [261]. Bluche *et al* [26] propose a multi-dimensional LSTM architecture associated with an attention mechanism to recognize handwritten text in paragraphs without explicit segmentation. Different from previous work, we follow the idea of *local* attention [162] which can be viewed as a blend between hard and soft attention. Our model focuses on the local context around the target states and avoids the expensive computation incurred in the soft attention. Thus, our model is scalable to images with long character sequences. Mnish *et al* [173] proposes a recurrent neural network to extract information from an image or video by adaptively selecting a sequence of regions or locations. Different from their work, we focus on handwritten text recognition and attend different regions during training and inference.

Reinforcement Learning. Reinforcement learning (RL) is to learn a policy network that determines certain actions under particular states. It is effective to optimize the sequential decision problems. Recently, several attempts have applied RL to computer vision tasks [30, 104, 108, 147, 253, 283]. Zhao *et al* [283] and Wu *et al* [253] explore deep RL to dynamically choose layers of CNNs during inference.A video object segmentation model [108] is proposed to learn object foreground-context regions by incorporating a reinforcement cutting-agent learning framework. In our work, we adopt a policy network to select context regions to attend according to the observed states during inference. Inspired by the BlockDrop model [253], we view our decision making process as an instantiation of associative reinforcement learning where all the decisions are taken in a single step.

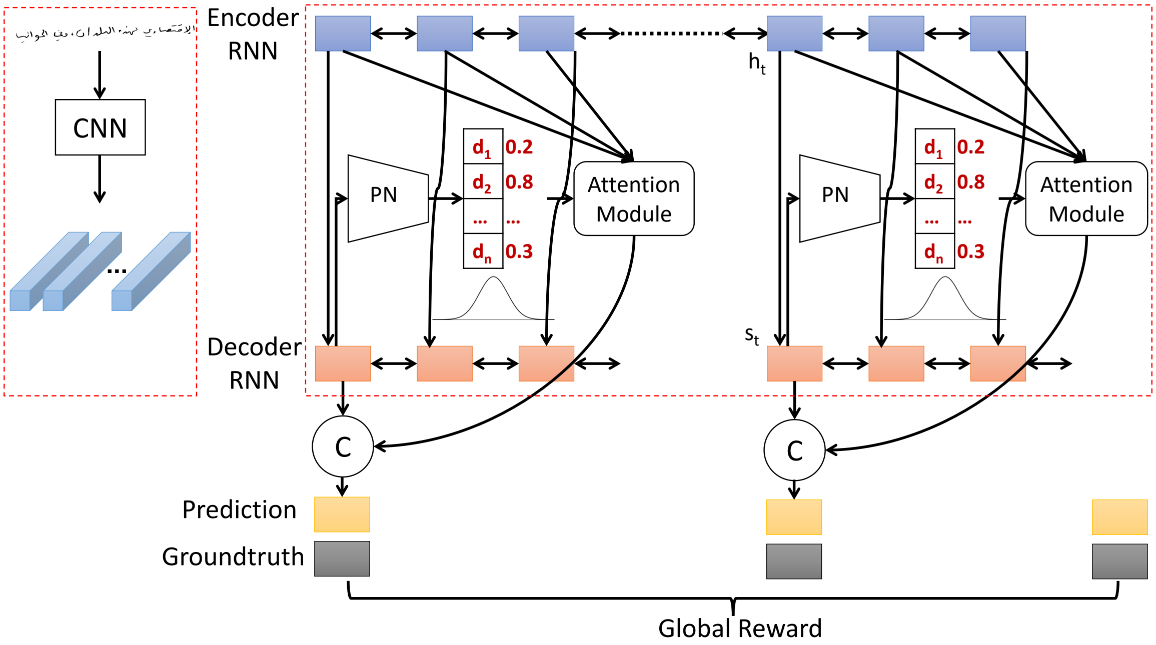


Figure 2.2: The framework of our proposed model. The policy network (PN) is trained to choose an optimal context length from the action set according to the observed state. The attention module then selectively attends to this context region and explicitly encode it into the local context. The context captured by LSTM and the local context are simultaneously taken into consideration during inference.

2.4 Context-aware Reinforced Agent

In offline handwritten text recognition tasks, the goal is to build a system which, given an image, produces a prediction of the image transcription. Our insight is that it is beneficial to simultaneously leverage both local context (as illustrated in Fig. 6.1) and global context. The key idea is that we adaptively select context region to attend during inference according to the observed states. Fig. 2.2 shows an overview of our framework.

Formally, given a dataset $S = \{(I, z)\}$, I is an image and z is the textual transcription. We take a raw image as input and encode it into a feature sequence s, where s_t is the state at time-step t. We train an adaptive context-aware reinforced agent to predict the context length of s_t. We then derive the expectation c_t within the window size by leveraging the soft attention mechanism. c_t is applied as the adaptive local context during inference. Details of the model are demonstrated in the following sections.

Visual Features Encoder

The visual features of an image are extracted from a fully convolutional neural network which consists of max-pooling layers. We model it using the CNN network [209] for OCR from images (Specification is given in Table 2.1). The network takes the raw inputs and produces feature maps that are robust and contain high-level descriptions of the input images. Suppose the feature maps are of size $D \times H \times W$, where D denotes the number of channels and H and W are the height and width of the feature maps.

According to the translation invariance property of CNN, each column of the feature maps corresponds to a local image region as the receptive field. The feature maps are then flattened into a sequence with a length of W, each of which has $D \times H$ dimensions. Specifically, each feature vector of the feature sequence is generated from left to right on the feature maps by column. We denote the the visual feature sequence as $v = (v_1, \ldots, v_W)$. We follow the same settings [209], and fix the height of each column H as a single pixel.

Restricted by the sizes of the receptive fields, the feature sequence leverages limited image contexts. We run a RNN over the feature sequence V to model the long-term dependencies within the sequence. Formally, a RNN is a parameterized function that recursively maps an input vector and a hidden state to a new hidden state. At time t, the hidden state is updated with an input v_t in the following manner: $h_t = RNN(h_{t-1}, v_t; \theta)$. For simplicity we will describe the model as a RNN, but all experiments use the BLSTM. We denote the encoded states from v as $h = h_1, \ldots, h_W$.

Conv	MaxPool	Conv	MaxPool	Conv	Conv	MaxPool	Conv	Conv	MaxPool	Conv
3×3	2×2	3×3	2×2	3×3	3×3	2×2	3×3	3×3	2×2	2×2
num: 64		num: 128		num: 256	num: 256		num: 512	num: 512		512
sh:1 sw:1	sh:2 sw:2	sh:1 sw:1	sh:2 sw:2	sh:1 sw:1	sh:1 sw:1	sh:2 sw:1	sh:1 sw:1	sh:1 sw:1	sh:2 sw:1	sh:1 sw:1
ph:1 pw:1	ph:0 pw:0	ph:1 pw:1	ph:0 pw:0	ph:1 pw:1	ph:1 pw:1	ph:0 pw:1	ph:1 pw:1	ph:1 pw:1	ph:0 pw:1	ph:0 pw:0

Table 2.1: The CNN architecture configuration.

Context Features Decoder

Considering the cursive and imprecise nature in the handwritten text recognition problem, our insight is that explicitly encoded local context (as illustrated in Fig. 6.1) is complementary to global context when determining observed states into characters. Given a feature sequence, learning the context region localization agent would result in a nearly continuous decision-making process. To simplify this problem, we discretize the context regions into an action set and leverage a policy network to make decisions in selecting appropriate context regions.

We introduce an adaptive context-aware agent to select and attend different context regions given states at different time-steps. We first leverage a BLSTM to extract higher level of abstractions from the encoder outputs s as $s_t = RNN(s_{t-1}, h_t; \theta)$.

Adaptive Context-aware Reinforced Agent. Our method is based on Q-learning, a kind of reinforcement learning, which focuses on how an agent ought to take actions so as to maximize the final reward. The Q-learning model consists of an *agent*, *states* and a set of *actions*.

We adopt s as the sequence *states*. The searching action set $\mathcal{A}$ contains different context lengths and is denoted as $\mathcal{A} = \{d_1, \ldots, d_n\}$, where n is the number of context lengths. For an input s_t, we design a policy network to learn the expected adaptive context-aware reinforced agent, which determines the action policy $a(s_t)$ according to the observed s_t. Both the *state* and *action* are finite and discrete to ensure a relatively small searching space. Given a $(s_t, a(s_t))$, we adopt the negative loss defined in Sec. 2.4 as our reward. Following the training strategy [253], we train the policy network to predict *all actions at once* which is different from the standard reinforcement learning algorithms and is essentially a single-step Markov Decision Process (MDP) given the input states. This can also be viewed as contextual bandit [122] or associative reinforcement learning [222].

Formally, given a sequence s, we define an action policy as a multinomial distribution:

$$\pi_W(a|s) = \prod_{t=1}^{T} p_t^{a_t}, \tag{2.1}$$

$$p = f_{pn}(s; W), \tag{2.2}$$

where f_{pn} denotes the *policy network* parameterized by weights W and p is the output of the network after the softmax function. We denote the probability of the corresponding action a_t at time-step t as $p_t^{a_t}$. To learn the optimal parameters of the policy network, we maximize the following expected reward:

$$J = \mathbb{E}_{a \sim \pi_W}[R(a)]. \tag{2.3}$$

To maximize Eqn. 2.3, we utilize policy gradient [222], one of the seminal policy search methods [60], to compute the gradients of J. The gradients can be derived as:

$$\nabla_W J = E[R(a) \nabla_W \log \pi_W(a|s)], \tag{2.4}$$

Where W denotes the parameters of the policy network. We approximate the expected gradient in Eqn. 2.4 with Monte-Carlo sampling using all samples in a mini-batch. To reduce variance [222] in these gradient estimates, we utilize a self-critical baseline $R(\tilde{u})$ as in [197] and Eqn. 2.4 can thus be rewritten as:

$$\nabla_W J = E[(R(a) - R(\tilde{a})) \nabla_W \log \pi_W(a|s)], \tag{2.5}$$

where $\tilde{a}$ is defined as the maximally probable configuration under the current policy. For example, $\tilde{a}$ is the action from $\mathcal{A}$ with the index of $argmax(p_t)$.

To further encourage exploration in policy searches, we adopt a parameter α to bound the distribution p and prevent it from saturating. The modified distribution p' can be formulated as:

$$p' = \alpha \cdot p + (1 - \alpha) \cdot (1 - p). \tag{2.6}$$

The modified distribution p' is applied when we sample the action policies.
Local Attention. Since not every time-step of the sequence is relevant for the prediction, the model should extract the salient parts. Our local attention mechanism selectively focuses on a small window of context. In concrete details, given a predicted window size D at time-step t, the source hidden states within the window are denoted as $h_{[t-\frac{D}{2}:t+\frac{D}{2}]}$. We follow past empirical work [162] and compute the attention weight vector as:

$$a^{att} = softmax(s_t^T W_a h_{[t-\frac{D}{2}:t+\frac{D}{2}]}), \tag{2.7}$$

where W_a is the projection vector which will be jointly trained with the model. Then the context at time-step t is defined as an expectation of s within the window of $[t - \frac{D}{2} : t + \frac{D}{2}]$:

$$c_t = \sum_i a_i^{att} h_i. \tag{2.8}$$

To take both the global context and explicitly encoded local context into consideration, we use the concatenation of s_t and c_t as the representation at time-step t.

In summary, our model works as follows: f_{pn} is used to decide which window size to attend conditioned on the input feature sequence. A prediction is generated by running a forward pass and we aim to maximize the total expected reward, or equivalently minimize the negative expected reward as our loss.

Transcription Layer

Transcription is a process of converting the per-frame predictions made by the decoder module into a label sequence. Mathematically, transcription procedure is to find the label sequence with the highest probability conditioned on the per-frame predictions.

In this section, We adopt Connectionist Temporal Classification (CTC) [82] layer to transform variable-width feature tensor into a conditional probability distribution over label sequence. The probability ignores the position where each per-frame prediction is located and avoids the labor of labeling positions of individual characters.

Formally, let $\mathcal{L}$ be the alphabet and $\hat{\mathcal{L}} = \mathcal{L} \cup \{-\}$ where $-$ is a blank character. Given an input image I, the generated predictions $\pi = \{\pi_1, \ldots, \pi_T\}$, where T is the sequence length and

$\pi \in \mathcal{R}^{\hat{\mathcal{L}}}$. The probability distribution over the alphabet $\hat{\mathcal{L}}$ is denoted as $y = \{y_1, ..., y_T\}$. We denote $y^t_{\pi_t}$ as the probability of generating label π_t at time-step t. The sequence π may contain blank characters and repeated labels. CTC defines a map function $\mathcal{B}$ which maps π to a concise representation l by removing blank characters and repeated labels (*e.g.*, hhee–ll-lo–=hello).

Thus, the probability of π is defined as $p(\pi|y) = \prod_{t=1}^{T} y^t_{\pi_t}$. The conditional probability of observing the output sequence l is then given as:

$$p(l|y) = \sum_{\pi:\mathcal{B}(\pi)=l} \log p(\pi|y). \tag{2.9}$$

Due to the exponentially large number of summation items, directly computing Eqn. 2.9 is computationally infeasible. While Eqn. 2.9 can be efficiently computed using the forward-backward algorithm [82].

2.5 Empirical Evaluation

Experimental Setup

In this section, we present our experiment setups by introducing the benchmarks, the experiment settings and evaluation metrics used for evaluation.

Datasets. We present results on the commonly used handwritten text recognition benchmarks. The datasets used are KHATT, IAM and RIMES, which contain images of handwritten Arabic, English and French, respectively. We use the same network for all experiments and no language specific information is needed except for the character set of each benchmark. A brief description of these benchmarks is as follows.

The KHATT [164] database is an offline handwritten text recognition database of cursive Arabic text documents. It contains $2,000$ paragraphs by $1,000$ writers. The paragraphs are segmented into a total number of $9,327$ lines. The database is provided with line level annotations and a standard data set splits.

The IAM [168] database is a handwritten text recognition database of mostly cursive English text documents. The training set comprises 747 documents ($6,482$ lines, $55,081$ words), the validation set 116 documents (976 lines, $8,895$ words) and the test set 336 documents ($2,915$ lines, $25,920$ words). The texts in this database typically contain 50 characters per line.

The RIMES [86] database contains more than $60,000$ words written by over $1,000$ authors in French. This database has several versions with each one a super-set of the previous one. We use the latest version presented in a ICDAR 2011 contest for our experiments.

Experiment settings. We follow the lexicon-based methods [7, 24, 70, 186] and use all the dataset words, both train and test sets, as the lexicon. The model's predictions are compared with the actual image transcriptions. To ease comparison to other algorithms, we report using the same measure commonly used in the respected benchmarks. On IAM and RIMES, we show our results using WER and CER measures. Whereas on KHATT, images are annotated at line level which makes the measure of WER infeasible. We report our results using CER calculated at sequence level.

Different character sets are used for the benchmarks. More specifically, the character set for IAM contains the lower and upper case Latin alphabet. Digits are not included as they are rarely used in this dataset. For RIMES, the character set contains the lower and upper case Latin alphabet, digits and accented letters. For KHATT, as the images are at line level, the character set contains the Arabic alphabet, comma, dot, space and unknown letters.

Evaluation protocols. We apply our model to the test set and compare the predicted transcription with the ground truth transcriptions. The performance can be measured by Word Error Rate (WER) and Character Error Rate (CER). WER is the ratio of the reading mistakes calculated at the word level. CER measures the Levenshtein distance normalized by the length of the ground-truth word. That is, we measure the total number of substitutions, insertions and deletions that would be required to turn the prediction sequence into the ground-truth one.

Implementation details. In our experiments, we binarize images by applying Otsu's method [180]. The heights of images are scaled to 32 and the widths are proportionally scaled with heights. The size of hidden states for encoder and decoder modules are set as 128. We implement the neural network using PyTorch. Parameter optimization is performed using the Adam algorithm [116] with a batch size of 32 and a learning rate of 0.01. To reduce the effects of "gradient exploding", we use a gradient clipping of 0.1 [182]. We insert batch normalization layer after each convolutional layer to accelerate the training process. We empirically set values of actions as $\mathcal{A} = \{1, 5, 10, 15, 20\}$. Training the network takes around 20min on KHATT dataset using a single GPU TITAN X.

Results and Discussion

To evaluate the effectiveness of our proposed algorithm, we conduct an extensive set of experiments on handwritten words recognition benchmarks. We also investigate the ablation studies on handwritten text lines recognition benchmarks.

Handwritten word recognition task. We compare to the state of the art on IAM and RIMES datasets in Table 2.2. Our model outperforms previous work by large margins on the handwritten words recognition benchmarks. Wigington *et al* [249] reports two results with/without data augmentation techniques on the test set. For a fair comparison, we compare the performance under the same experiment settings by leveraging the training set only. As Shi *et al* [209] is closely related to our work, we report the performance on two benchmarks. While their work is focused on scene text recognition, it is still competitive compared to other previous work. Our model outperforms Shi *et al* [209] which indicates the adaptive context-aware reinforced agent can help with recognizing handwritten words.

Database	IAM		RIMES	
Model	WER	CER	WER	CER
Boquera *et al* [70]	15.50	6.90	-	-
Telecom ParisTech [85]	-	-	24.88	-
IRISA [85]	-	-	21.41	-
Jouve [85]	-	-	12.53	-
Kozielski *et al* [118]	13.30	5.10	13.70	4.60
Almazan *et al* [7]	20.01	11.27	-	-
Messina and Kermorvant [170]	19.40	-	13.30	-
Pham *et al* [184]	13.60	5.10	12.30	3.30
Bluche *et al* [24]	20.50	-	9.2	-
Doetsch *et al* [64]	12.20	4.70	12.90	4.30
Bluche *et al* [25]	11.90	4.90	11.80	3.70
Shi *et al* [209]	6.74	3.75	4.23	2.10
Menasri *et al* (combined) [169]	-	-	4.75	-
Poznanski *et al* [186]	6.45	3.44	3.90	1.90
Wigington *et al* [249]	7.18	3.93	3.84	1.82
Our work	**5.45**	**3.10**	**2.97**	**1.45**

Table 2.2: Comparison to previous methods on IAM and RIMES (ICDAR2011) datasets. Our model achieves the state-of-the-art performance by large margins on both benchmarks. All numbers are in percent.

Handwritten line recognition task. To test the scalability to long sequences (*e.g.*, 60 characters per sequence in KHATT dataset), we compare our model to the state-of-the-art algorithms on IAM and KHATT benchmarks. Our models are trained and evaluated using full lines. The comparisons are as shown in Table 2.3. We report the performance of Shi *et al*'s work [209], as it is closely related to our work and can be viewed as a baseline. Our model lowers the error rate by 1.7% compared to the baseline model. On IAM dataset, we compare our model to Bluche *et al*'s work which achieves remarkable performance on multi-line handwritten recognition [26]. Our model outperforms their work on both line and isolated word recognition.

Ablation studies. To investigate the impact of our proposed model, we conduct an extensive set of experiments. The first experiment is to validate if local attention mechanism outperforms global attention over the full sequence. As shown in Table 2.3, the global attention performs worse on both benchmarks. One possible reason is that unlike other tasks (*e.g.*, machine translation), global attention introduces more noise when dealing with long sequences due to the imprecise nature of handwriting. We then replace the adaptive context-aware reinforced agent with a single fixed-size. The window size is empirically set as 9, the median value of our action sets. This modified model performs better than the baseline while consistently worse than our proposed model on both benchmarks.

Database	IAM	KHATT
Model	CER	CER
Shi *et al* [209]	6.20	8.65
Bluche *et al* (w/o attention) [26]	6.60	-
Bluche *et al* (w/ attention) [26]	7.00	-
Our work (w/ GA)	8.35	10.20
Our work (w/ fixed-size LA)	5.91	7.62
Our work (full model)	**5.15**	**6.93**

Table 2.3: Comparison to previous methods and ablation studies on IAM and KHATT datasets. Our experiments are conducted on full lines instead of isolated words. All numbers are in percent. GA: global attention, LA: local attention.

2.6 Summary

In this paper, we have made a pioneer effort to formulate handwritten text recognition in a reinforcement learning framework and propose a novel adaptive context-aware reinforced agent to tackle this problem. The proposed method can generalize well from isolated word recognition to full lines recognition. Comprehensive experiments on commonly used benchmark datasets demonstrate the effectiveness of the proposed method. In the future, we plan to extend this method to multi-lines and paragraphs recognition without pre-segmentation.

Chapter 3

Few-Shot Image Classification with Knowledge-Guided Data Augmentation

In this chapter, we address the challenge of data scarcity in few-shot image classification tasks by incorporating structured human prior knowledge into data augmentation strategies. The assumption that similar objects exhibit similar behaviors is formalized to create synthetic samples, enabling better generalization from limited data. This approach highlights the potential of leveraging human intuition to enhance classification models in data-constrained scenarios.

3.1 Overview

Learning to hallucinate additional examples has recently been shown as a promising direction to address few-shot learning tasks. This work investigates two important yet overlooked natural supervision signals for guiding the hallucination process – (i) extrinsic: classifiers trained on hallucinated examples should be close to strong classifiers that would be learned from a large amount of real examples; and (ii) intrinsic: clusters of hallucinated and real examples belonging to the same class should be pulled together, while simultaneously pushing apart clusters of hallucinated and real examples from different classes. We achieve (i) by introducing an additional mentor model on data-abundant base classes for directing the hallucinator, and achieve (ii) by performing contrastive learning between hallucinated and real examples. As a general, model-agnostic framework, our dual mentor- and self-directed (DMAS) hallucinator significantly improves few-shot learning performance on widely-used benchmarks in various scenarios.

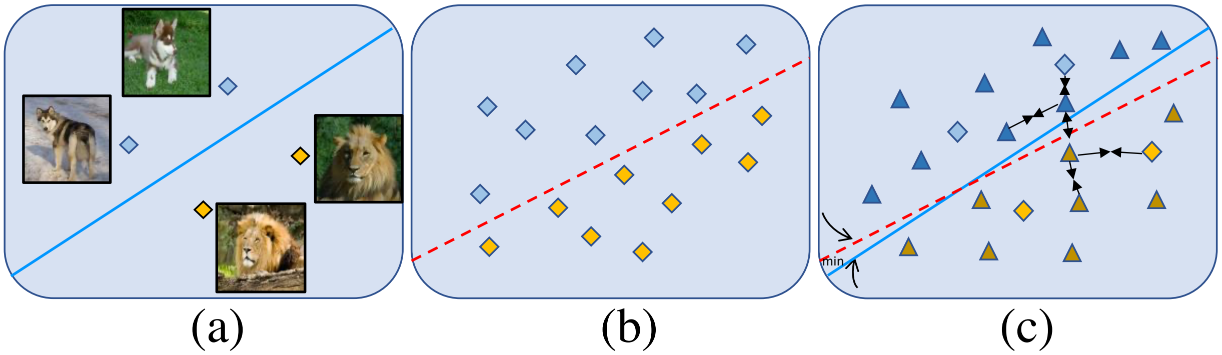


Figure 3.1: Learning a hallucinator to generate useful examples for few-shot learning through extrinsic and intrinsic supervision.

3.2 Motivation

To alleviate the reliance on large, labeled datasets for learning deep models, few-shot learning has attracted increasing attention, with the goal of learning novel concepts from one, or only a few, annotated examples [72, 73, 213, 233, 242]. Existing work tries to solve this problem from the perspective of meta-learning [20, 203, 225], which is motivated by the human ability to leverage prior experiences when tackling a new task. Unlike the standard machine learning paradigm, where a model is trained on a set of examples, meta-learning is performed on a set of "simulated" tasks, each consisting of its own support and query sets [233]. The support set is used as the few-shot training data for the leaner, and the query set is used as the test data to evaluate the leaner's quality. By sampling small support and query sets from a large collection of labeled examples of base classes, meta-learning based approaches learn to extract task-agnostic knowledge, and apply it to a new few-shot learning task of novel classes.

One notable type of task-agnostic (or meta) knowledge comes from the shared mechanism of data augmentation or *hallucination* across categories [74, 204, 244, 280]. Since synthesizing raw images is often challenging or sometimes unnecessary, recent work has instead focused on hallucinating examples in a *learned feature space* [74, 204, 244, 255, 275, 280, 281]. This can be achieved by, for example, integrating a "hallucinator" module into a meta-learning framework, where it generates hallucinated examples guided by real ones from the support set [244]. The hallucinator captures the *intra-class variation* shared across categories, which generalizes to unseen classes. The learner then uses an augmented training set, which includes both the real and the hallucinated examples to learn classifiers. The hallucinator is meta-trained end-to-end with the learner, through back-propagating a classification loss based on ground-truth labels of query data.

Despite the success of prior approaches, we argue that solely using the classification loss on the *small query set* as supervision is insufficient to adjust the hallucinator to produce effective samples in the few-shot regime. Therefore, the performance of the classifiers trained on hallucinated examples is still substantially inferior to that of the classifiers trained on real examples [57, 210]. To overcome this challenge, *our key insight* is that there are *two important yet under-explored natural signals* for guiding the data generation process – *extrinsic and intrinsic supervision*. This work explores how to leverage such supervision to enable hallucinating examples in a way that helps the classification algorithm learn better classifiers.

The first source of supervision is an *extrinsic signal from large-sample learning*. As illustrated in Figure 3.1, to be most helpful as a hallucinator, a classifier trained on the hallucinated examples (which are generated from a small support set of real samples) is expected to be *close* to a strong classifier that would be trained on a large amount of real examples . This extrinsic signal from large-sample learning is a natural source of supervision for few-shot learning, but it has been largely overlooked in prior work. While we have very little data on novel classes, we *do have a large number of real examples on base classes*. Therefore, on base classes we introduce a "mentor" model, which is a strong classifier pre-trained on all the available large amount of real examples. Correspondingly, the classifier trained on hallucinated examples along with few real support examples becomes the "student."

We now minimize the discrepancy between the student and mentor classifiers. A straightforward approach would be minimizing the distance between the two classifiers in the parameter space [33, 242, 243], which tends to be difficult and noisy due to the lack of suitable metrics. Hence, we instead encourage the output predictions from the student classifier (*e.g.*, the distribution of class probabilities) to be similar to those predicted by the mentor on the query set. This way of learning is reminiscent of knowledge distillation [92]. By doing so, the hallucinator explicitly learns how to produce examples that enable the student classifier to mimic the behavior of the mentor. Note that *the student-mentor pairs are only used for meta-training on base classes; there are no mentor classifiers for meta-testing on novel classes.*

In practice, the student and mentor classifiers could be quite different from each other at the beginning of the training, if the mentor is produced by a large amount of real examples while the student has access to only few real examples. To address this issue, we propose *a progressive guidance scheme* inspired by curriculum learning [21], and explore two *dual* directions – (1) we start with a mentor and a student, both trained on a small number of real examples, and we gradually *strengthen* the mentor by re-training it with increasing number of real examples; and (2) we start with a mentor and a student, both trained on a large number of real examples, and we gradually *weaken* the student by removing its real examples. During both of the processes, the hallucinator is also trained progressively.

The second source of supervision is the *intrinsic label consistency between hallucinated and real examples*. As illustrated in Figure 3.1, hallucinated and real examples belonging to the same class should be pulled together, while simultaneously pushing apart clusters of hallucinated and real examples from different classes. However, without appropriate constraints, the hallucinated examples might be noisy and spread over across class boundaries (*e.g.*, a hallucinated dog example resides within the cat cluster). To this end, we formulate the problem as *supervised contrastive learning*, inspired by recent progress on self-supervised learning [43, 90, 112, 252]. We treat hallucinated and real examples as different views of the data, and generate the positive and negative pairs correspondingly. For example, the positives are drawn from both hallucinated and real samples of the same class. Note that different from conventional contrastive learning that learns an embedding space (where the data augmentation is pre-defined), we use the contrastive loss to self-direct the hallucinated examples in the right class cluster or manifold (where is the feature space is pre-trained).

As shown in Figure 3.1, during meta-training, we sample a few-shot task (*e.g.*, 2-way 2-shot classification) on **base classes** (Fig. 3.1a). **Extrinsic supervision:** The desired classifier for this task is the (dashed) one that would be learned from a large set of real examples (Fig. 3.1b). We *explicitly* introduce this strong classifier as "mentor" (abundant examples are available for base classes). We then learn the hallucinator in a way that minimizes the discrepancy between the (solid) "student" classifier (trained on hallucinated examples together with the few real examples) and the (dashed) mentor classifier (Fig. 3.1c). **Intrinsic supervision:** Through contrastive learning, clusters of hallucinated and real examples belonging to the same class are pulled together (→←), while simultaneously pushing apart (↔) clusters of hallucinated and real examples from different classes (Fig. 3.1c). During meta-testing, we use the *meta-trained, fixed* hallucinator to generate additional examples as augmentation for learning classifiers on **novel classes**. Real examples as light diamonds, hallucinated examples as dark triangles, and classifiers as solid or dashed lines.

Our contributions are three-fold. (1) By jointly leveraging the *complementary* extrinsic and intrinsic supervision, we develop a general meta-learning with hallucination framework. (2) We not only extract shared knowledge across a collection of few-shot learning tasks, similar to most existing meta-learning methods, but also *progressively* exploit extrinsic knowledge in *large-sample models trained on base classes as mentor* to guide hallucination and few-shot learning. (3) Through a contrastive learning process, the hallucinated examples are self-directed to maintain the intrinsic label consistency with real examples. Our dual mentor- and self-directed (DMAS) hallucinator is *model-agnostic*, which can *generate data in different feature spaces and can be combined with different classification models* to consistently boost their few-shot learning performance on a variety of benchmarks, including ImageNet1K [88, 244], *mini*ImageNet [193, 233], *tiered*ImageNet [195], and CUB [235].

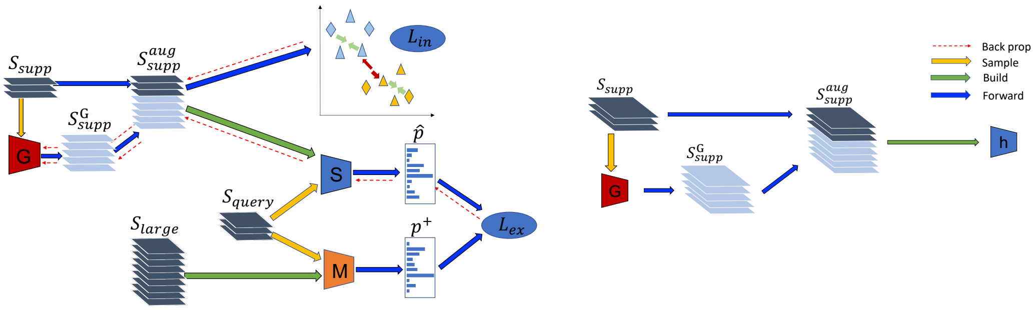
3.3 Prior Work

Generative Models. Generative models have recently shown great potential as a way of data augmentation for few-shot learning [10, 74, 244, 280] and semi-supervised learning [57], but the improvement of recognition performance is still limited [210]. The generation can be performed either in image space [50] or in a pre-trained feature space [88], by using an auto-encoder architecture [204], GAN-like generator [244], or the combination of GANs and auto-encoders [254, 255]. Our work is independent of these different types of generators, and we focus primarily on how to train the generator to improve its use for recognition tasks by leveraging large amounts of auxiliary data and self-supervision.

Few-Shot Learning and Meta-Learning. Meta-learning, or the ability to *learn to learn* [225], is a powerful framework for tackling the problem of learning with limited data. Most of modern approaches fall into one of the categories between optimization and metric learning based methods. Optimization based methods learn how to do fast adaptation to novel tasks, by learning appropriate parameter updates [193] or a general initialization [73]. Adaptation could be done in the original feature space [11, 12, 73] or in an embedded space [199]. Prior work on few-shot domain adaptation [113, 200] learns how to balance cross-domain clustering that is domain invariant. Metric learning methods focus on learning a similarity metric [117]. Several distance functions have been explored, from the Euclidean distance [6, 213] and the cosine distance [44, 69, 76] to more complex parametric functions and metrics [142, 221, 233, 273], or using an additional task-specific metric [179]. Most methods often treat each category separately without considering the relations between them. Graph neural networks are thus introduced to leverage those relations [77, 114, 202]. To conduct meta-learning more effectively, recent approaches often first compute a set of features of the images using a trained feature extractor network. Given that high-dimensional features have better modeling capacity but are computationally expensive to work with, each meta-learning task is then formulated as a convex optimization problem and solved in its low-dimensional dual space [22, 126]. Our hallucinator component is generic and can be integrated into different meta-learning methods.

Teacher-Student Networks. Learning a model under the guidance of a teacher or mentor model has been widely used for model compression. Compressing one cumbersome or several models into a smaller model is a classic idea [28, 65] and has been popularized by the distillation formulation in [92]. Recent work focuses on advanced techniques to guide the distillation process [3, 171, 262] and its applications to practical problems, such as object detection [247, 259] and distributed machine learning [8]. In addition, knowledge distillation has been extended to address other tasks, including multi-task learning [224] and continual learning [146, 205]. To the best of our knowledge, our work is the first to introduce a mentor network for learning recognition task oriented generative models. Importantly, different from existing work that addresses models of different capacity, we consider *models of the same capacity but trained on real or synthetic data*.

Contrastive Learning. Powerful self-supervised representation learning approaches have recently been developed in image domain via manually specified pretext tasks. Examples include auto-encoding methods which leverage contexts [183], channels [279], and colors [278] to recover the input under some corruption. Some pretext tasks form pseudo-labels by relative patch locations [63], image rotations [78], and jigsaw puzzles [176]. These pretext tasks are collected under the umbrella of the contrastive learning framework, which maintains the relative consistency between the representations of an image and its augmented views [43, 46, 84, 90, 177, 226, 252, 267, 271]. In our work, we treat hallucinated and real examples as different views of the data and use the contrastive loss to self-direct the hallucinated examples in the right class cluster or manifold.



(a) Meta-training on base classes

(b) Meta-testing on novel classes

Figure 3.2: Overview of our dual mentor- and self-directed hallucinator "DMAS," learned through extrinsic and intrinsic supervision. Real examples as diamonds, hallucinated examples as triangles.

3.4 Dual Mentor- and Self-Directed Hallucinator

Few-Shot Learning Setting. We are given a set of base categories $\mathcal{C}_{\text{base}}$ and a set of novel categories $\mathcal{C}_{\text{novel}}$, where $\mathcal{C}_{\text{base}} \cap \mathcal{C}_{\text{novel}} = \emptyset$. We have a base dataset $\mathcal{D}_{\text{base}}$ with a large amount of annotated training examples per class and a novel dataset $\mathcal{D}_{\text{novel}}$ with only few annotated training examples per class. Few-shot learning aims to learn a good classification model h for $\mathcal{C}_{\text{novel}}$ based on the small dataset $\mathcal{D}_{\text{novel}}$. Recent work achieves this through a meta-learning procedure [233], which learns from a collection of sampled few-shot classification tasks on $\mathcal{C}_{\text{base}}$. Given a set of categories $\mathcal{C}$ and a set of data $\mathcal{D}$, an m-way k-shot task is composed of a subset $\mathcal{C}_{\text{sub}}$ of m categories from $\mathcal{C}$, a support (training) set $\mathcal{S}_{\text{supp}}$ of k examples from $\mathcal{D}$ for each class in $\mathcal{C}_{\text{sub}}$, and a query (test) set $\mathcal{S}_{\text{query}}$ of one or few examples from $\mathcal{D}$ for each class in $\mathcal{C}_{\text{sub}}$. Meta-learning is performed in two phases as follows.

During *meta-training*, a classifier learns from a collection of m-way k-shot tasks sampled from $\mathcal{C}_{\text{base}}$ and $\mathcal{D}_{\text{base}}$. While our work is agnostic to different classification models, here we take a simple cosine classifier [44] as an example – a variant of prototypical networks [213] which uses the cosine instead of the standard Euclidean distance function. In each iteration, we compute a prototype representation for each class in $\mathcal{C}_{\text{sub}}$. Each example is fed to an embedding function f_θ with learnable parameters θ. The prototype of class c is the mean of the outputs through f_θ of examples from c in $\mathcal{S}_{\text{supp}}$. We then feed the examples in $\mathcal{S}_{\text{query}}$ to the classifier and update the parameters θ. During *meta-testing*, we use the same approach and build our previously meta-learned classifier with one unique m-way k-shot task, using $\mathcal{C}_{\text{novel}}$ instead of $\mathcal{C}_{\text{base}}$ and $\mathcal{D}_{\text{novel}}$ instead of $\mathcal{D}_{\text{base}}$. We evaluate the final classifier on unseen examples with labels from $\mathcal{C}_{\text{novel}}$.

Meta-Learning with Hallucination. Incorporating a generative model which produces additional examples for data augmentation has been shown to facilitate meta-learning [74, 204, 244]. While our approach does not rely on specific types of generative models, here we focus on the *feature hallucinator* in [244], due to its simplicity and state-of-the-art performance, which is implemented as a light-weight multi-layer perceptron (MLP) module. The hallucinator is a function $G(x, z; w) : \mathcal{R}^{d+d_{\text{noise}}} \to \mathcal{R}^d$ that produces examples in a pre-trained feature space of dimension d, where x is the feature vector of a real example, z is a random noise vector of dimension d_{noise} sampled from a Gaussian distribution, and w is the parameters of G. The hallucinated example $G(x, z; w)$ is of the same category as x.

Now the procedure of meta-learning integrated with the hallucinator G is illustrated in Figure 3.2. During each iteration of meta-training, the support set $\mathcal{S}_{\text{supp}}$ is first augmented by a generated set $\mathcal{S}^G_{\text{supp}}$. Specifically, for each class y, we sample $k^{\text{gen}}_{\text{train}}$ examples (x, y) in $\mathcal{S}_{\text{supp}}$, sample associated random noise vectors z, and then add (x', y) to $\mathcal{S}^G_{\text{supp}}$, where $x' = G(x, z; w)$. Our final support training set is $\mathcal{S}^{\text{aug}}_{\text{supp}} = \mathcal{S}_{\text{supp}} \cup \mathcal{S}^G_{\text{supp}}$. As long as G is differentiable with respect to the generated set $\mathcal{S}^G_{\text{supp}}$, the gradients of the final classification loss on $\mathcal{S}_{\text{query}}$ can be back-propagated into G to produce useful hallucinated examples. Through meta-training over a large amount of iterations, the hallucinator learns to capture shared modes of variation across different classes and can thus generalize to unseen classes. During meta-testing, we use the learned G to generate additional examples for recognizing categories in $\mathcal{C}_{\text{novel}}$.

Hallucination with Extrinsic Guidance from Mentor. The end-to-end optimization of the classification loss enables the hallucinator to produce useful examples in the few-shot regime. However, since the classification loss is computed on *small query sets*, such supervision solely is insufficient to adjust the hallucinator to produce *discriminative* examples that most contribute to formulating classifier decision boundaries. Hence, the resulting classifier trained on the hallucinated examples could be still far away from the desired classifier that would be learned from a large set of real examples. This makes it critical to close the gap between these two classifiers. In fact, during meta-training, a large amount of annotated examples are already available for the base categories $\mathcal{C}_{\text{base}}$, which allows us to explicitly obtain the classifier trained on a large set of examples and use it to guide the learning of the hallucinator.

Formally, we treat the classifier trained on the augmented set of the hallucinated examples and the few support examples as a *student model*, and we treat the classifier trained on a large set of real base examples as a *mentor model*. Our goal then is to learn the hallucinator by minimizing the discrepancy between the student classifier and its mentor model. While a naïve approach would be to directly characterize the difference between their model parameters, it turns out to be challenging due to the high dimensionality of the parameter space. Inspired by the teacher-student network [92], we instead enforce the student to mimic the distribution of class probabilities predicted by the mentor network, which can be viewed as a way of regularization to improve the generalization performance of the student model [172].

As shown in Figure 3.2, meta-training the hallucinator G is conducted in the following way. We first sample a *large* set of examples $\mathcal{S}_{\text{large}}$ with k_{large} examples per class in $\mathcal{C}_{\text{base}}$ and train a mentor classifier using all the examples in $\mathcal{S}_{\text{large}}$. During each iteration of meta-training, we augment $\mathcal{S}_{\text{supp}}$ by generating new examples using the hallucinator G. We train the student classifier on $\mathcal{S}_{\text{supp}}^{\text{aug}}$ through the knowledge distillation loss function in [92]:

$$\mathcal{L}_{\text{ex}}(s, m, y) = \mathcal{L}_{\text{CE}}(\sigma(s), e_y) + \alpha\tau_1^2\mathcal{L}_{\text{CE}}(\sigma(\frac{s}{\tau_1}), \sigma(\frac{m}{\tau_1})), \tag{3.1}$$

which consists of a standard cross-entropy loss (the first term) and an additional component that measures the difference between student and mentor outputs (the second term). s and m are the logits produced by the student and the mentor, respectively, for a test example of label y in $\mathcal{S}_{\text{query}}$. σ denotes the softmax function, $\mathcal{L}_{\text{CE}}$ denotes the cross-entropy loss, e_y is the one-hot encoding of y, and α is a trade-off hyper-parameter that balances the two terms. Note that $\tau_1 > 0$ is a critical *learnable* parameter called *temperature*, which smooths the probability distribution produced by the mentor and makes the corresponding decision boundary easier to learn for the student than the original one.

Self-Directed Learning with Intrinsic Label Consistency. While the hallucinated examples directed by the extrinsic mentor are useful, without other constraints they might spread over across class boundaries and thus be noisy. Inspired by supervised contrastive learning [112], we enforce intrinsic label consistency between hallucinated examples and real examples.

Formally, suppose we sample N real examples per mini-batch and generate M hallucinated examples, resulting in a batch $\mathcal{I}$ of $M + N$ examples. Given an anchor example x_i, $P(i)$ is the set of indices of all positives in the batch distinct from i and $A(i) \equiv I\backslash\{i\}$. The supervised contrastive loss is defined as $\mathcal{L}_{\text{in}} = \frac{1}{M+N}\sum_{i=1}^{M+N}\mathcal{L}_i$ and

$$\mathcal{L}_i = -\frac{1}{|P(i)|}\sum_{p\in P(i)} \log \frac{exp(x_i \cdot x_p/\tau_2)}{\sum_{a\in A(i)} exp(x_i \cdot x_a/\tau_2)}, \tag{3.2}$$

where $\tau_2 > 0$ is a temperature parameter and $|P(i)|$ is its cardinality. This loss allows the real and hallucinated examples from the same classes to attract mutually, while they repel the other examples from different classes in the mini-batch.

Thus, our dual mentor- and self-directed hallucinator can be derived from Eqn. 3.1 and Eqn. 3.2 as

$$\mathcal{L} = \mathcal{L}_{\text{ex}} + \beta\mathcal{L}_{\text{in}}, \tag{3.3}$$

where β is a trade-off hyper-parameter that balances the two terms. Minimizing Eqn. 3.3 over $\mathcal{S}_{\text{query}}$ thus guides the hallucinator towards producing useful examples that help the student classifier recover the decision boundary from the mentor model.

Progressive Guidance from Mentor Model

Under the framework of meta-learning with extrinsic guidance, a straightforward way is to build the mentor model by using k_{large} as large as possible (potentially the full set of $\mathcal{D}_{\text{base}}$) and keep it fixed, and to train the hallucinator and student classifier using only few real examples. By doing so, however, we face the problem that the decision boundaries obtained by those two models could be very far from each other at the beginning of the training, making the learning of the hallucinator difficult. To address this issue, we perform the learning process in a progressive manner with *varied* number of real examples. We start with a mentor and a student which have access to a *not too different* number of real examples, and then progressively change the number of examples, so that the decision boundaries transform in a smooth manner. Concretely, this can be achieved in the following two *dual* directions.

Progressive Guidance by Strengthening the Mentor. In this setting, both the student and the mentor start with a small number of real examples. However, the number of real examples for the mentor *gradually increases over the training*. The objective for the hallucinator then is to learn to generate additional examples so that its corresponding student can always *match* the performance of the mentor, whenever the mentor is re-trained with more samples and becomes stronger. More specifically, during meta-training, the support set $\mathcal{S}_{\text{supp}}$ of each few-shot task is composed of very few examples per class, k_{train}, as in regular meta-training. At the beginning, we sample $\mathcal{S}_{\text{large}}$, with k_{large} being set to the value of k_{train}. We then progressively sample new real examples in the same amount for each class and add them into $\mathcal{S}_{\text{large}}$. k_{large} grows from k_{train} to $k_{\max}$ in a linear or logarithmic scale, where $k_{\max}$ is the maximum available number of examples per class in $\mathcal{D}_{\text{base}}$. We re-train the mentor model every time we add new examples.

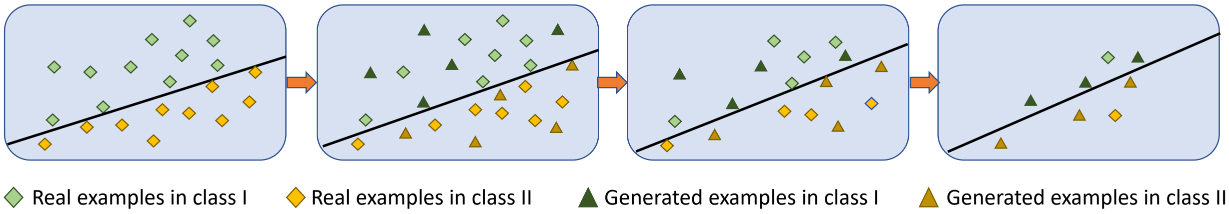

◇ Real examples in class I
◆ Real examples in class II
▲ Generated examples in class I
▲ Generated examples in class II

Figure 3.3: Illustration of progressive guidance by weakening the student classifier in the case of recognizing two classes.

Progressive Guidance by Weakening the Student. In this setting, both the student and the mentor start with a large number of real examples. However, we *gradually remove* the real examples for the student over the training. The objective for the hallucinator then is to learn to generate the missing examples based on the remaining real examples. This allows the student to *preserve or stabilize* the original decision boundary formulated by the large set of examples (*i.e.*, the mentor boundary), when the student has access to less real examples and becomes weaker. More specifically, during meta-training, the support set $\mathcal{S}_{\text{supp}}$ of each "few-shot" task is composed of a large number of examples per class, unlike regular meta-training. This number of examples per class in $\mathcal{S}_{\text{supp}}$, k_{train}, decreases in a linear or logarithmic scale, until it reaches a small value.

As shown in Figure 3.3, we start with a large number of real examples for both the student and the mentor, and learn the corresponding mentor model (the leftmost image). We then *gradually remove* the real examples for the student over the training. The hallucinator learns to generate additional examples based on the remaining real examples to *preserve* the mentor decision boundary (the middle two and rightmost images).

3.5 Empirical Evaluation

We now present experiments to evaluate our dual mentor- and self-directed (DMAS) hallucinator on few-shot classification, and study the effect of progressive guidance from extrinsic and intrinsic supervision. Since DMAS is agnostic to the choice of classification models, we validate its generalizability to different types of features and various meta-learning models. In particular, we focus on simple cosine classifiers, which have been recently shown to achieve very competitive few-shot performance [44].

Datasets. We evaluate on four widely-used datasets: (1) *mini*ImageNet [193, 233], with 64, 16, and 20 classes for meta-training, meta-validation, and meta-testing, respectively; (2) *tiered*ImageNet [195], with 20, 6, and 8 super-classes for meta-training, meta-validation, and

Method	Backbone	*mini*ImageNet		*tiered*ImageNet	
		k=1	5	k=1	5
Cosine Classifier [44]	ResNet12	55.43 ± 0.81	77.18 ± 0.61	61.49 ± 0.91	82.37 ± 0.67
TADAM [179]	ResNet12	58.50 ± 0.30	76.70 ± 0.30	–	–
ECM [194]	ResNet12	59.00 ± –	77.46 ± –	63.99 ± –	81.97 ± –
TPN [156]	ResNet12	59.46 ± –	75.65 ± –	59.91 ± 0.94	73.30 ± 0.75
PPA [188]	WRN-28-10	59.60 ± 0.41	73.74 ± 0.19	–	–
ProtoNet [213]	ResNet12	60.37 ± 0.83	78.02 ± 0.57	65.65 ± 0.92	83.40 ± 0.65
wDAE-GNN [77]	WRN-28-10	61.07 ± 0.15	76.75 ± 0.11	68.18 ± 0.16	83.09 ± 0.12
MTL [218]	ResNet12	61.20 ± 1.80	75.50 ± 0.80	–	–
LEO [199]	WRN-28-10	61.76 ± 0.08	77.59 ± 0.12	66.33 ± 0.05	81.44 ± 0.09
DC [148]	ResNet12	62.53 ± 0.19	79.77 ± 0.19	–	–
MetaOptNet [126]	ResNet12	62.64 ± 0.82	78.63 ± 0.46	65.99 ± 0.72	81.56 ± 0.53
FEAT [266]	ResNet24	62.96 ± 0.20	78.49 ± 0.15	–	–
MatchingNet [233]	ResNet12	63.08 ± 0.80	75.99 ± 0.60	68.50 ± 0.92	80.60 ± 0.71
CTM [131]	ResNet18	64.12 ± 0.82	80.51 ± 0.13	68.41 ± 0.39	84.28 ± 1.73
RFS [227]	ResNet12	64.82 ± 0.60	82.14 ± 0.43	71.52 ± 0.69	86.03 ± 0.49
DeepEMD [273]	ResNet12	65.91 ± 0.82	82.41 ± 0.56	71.16 ± 0.87	86.03 ± 0.58
DMAS (Ours)	ResNet12	**67.42 ± 0.28**	**83.74 ± 0.20**	**73.54 ± 0.73**	**86.27 ± 0.47**

(a) Test accuracy (%) on the novel classes for *mini*ImageNet and *tiered*ImageNet. '±' indicates 95% confidence intervals over tasks.

Method	Backbone	k=1	5
ProtoNet [213]	ResNet12	66.09 ± 0.92	82.50 ± 0.58
RelationNet [44, 221]	ResNet34	66.20 ± 0.99	82.30 ± 0.58
DEML [**?**]	ResNet50	66.95 ± 1.06	77.11 ± 0.78
MAML [44]	ResNet34	67.28 ± 1.08	83.47 ± 0.59
Cosine Classifier [44]	ResNet12	67.30 ± 0.86	84.75 ± 0.60
MatchingNet [233]	ResNet12	71.87 ± 0.85	85.08 ± 0.57
DeepEMD [273]	ResNet12	75.65 ± 0.83	88.69 ± 0.50
DMAS (Ours)	ResNet12	**78.47 ± 0.62**	**90.67 ± 0.39**

(b) Test accuracy (%) on the novel classes for CUB. '±' indicates 95% confidence intervals over tasks.

Method	Backbone	k=1	2	5	10
ProtoNet [213]	ResNet10	39.3	54.4	66.3	71.2
ProtoNet *Gen* [244]	ResNet10	45.0	55.9	67.3	73.0
MatchingNet [233]	ResNet10	43.6	54.0	66.0	72.5
Logistic regression [88]	ResNet10	38.4	51.1	64.8	71.6
Logistic regression *Analogies* [88]	ResNet10	40.7	50.8	62.0	69.3
Prototype Matching Net *Gen* [244]	ResNet10	45.8	57.8	69.0	74.3
Cosine Att. Weight [76]	ResNet10	46.0	57.5	69.1	74.8
DMAS (Ours)	ResNet10	**46.5**	**58.3**	**69.7**	**75.1**

(c) Top-5 accuracy (%) for **311-way** novel-class classification on ImageNet1K. **The 95% confidence intervals for all number are of the order of 0.2%**.

Table 3.1: Comparisons with state of the art on four widely-benchmarked few-shot classification datasets. *With simple cosine classifiers*, our DMAS significantly and consistently outperforms all the baselines (including sophisticated classification models) across the board.

meta-testing, respectively; (3) ImageNet1K [88, 244], with 193 base and 300 novel classes for cross-validation and 196 base and 311 novel classes for evaluation; (4) Caltech-UCSD Birds-200-2011 (CUB) [235, 266], with 100, 50, and 50 classes for meta-training, meta-validation, and meta-testing, respectively.

Implementation Details. For a fair comparison with previous work, we employ ResNet10 as our model backbone for ImageNet1K [244] and ResNet12 as our model backbone for the other three datasets [273]. As is commonly implemented in the state-of-the-art work, we follow the feature pre-training step [273]. We first train a convolutional network based feature extractor on the base classes. Then we extract and save these features to disk, and use these pre-computed features as inputs for meta-learning. We follow the feature hallucinator architecture in [244] and use a three layer MLP with ReLU as the activation. The embedding function f_θ of our cosine classifier is a two layer MLP.

During progressive guidance by weakening the student, we start training the mentor with $k_{\text{large}} = 256$, and we decrease the number to 1 in a logarithmic scale over $12,000$ iterations. We initialize the learnable parameters including the temperature τ_1 to 7, the scale factor of the cosine distance to 75, and the temperature τ_2 to 0.07. As the performance is not sensitive to trade-off hyper-parameters α and β, we empirically set them to 5 and 1, respectively. The number of hallucinated examples is a hype-parameter ranging from $2 - 10$. The saturation point of hallucinated examples on improving performance is typically 6. For ImageNet1K, we follow the settings in [244] and average over 5 pre-determined k-shot (*i.e.*, $k = 1, 2, 5, 10$) tasks. We report the mean top-5 accuracy and the 95% confidence intervals for all number are of the order of 0.2%. For the other datasets, we average over $1,000$ randomly sampled tasks and report the accuracies and the 95% confidence intervals.

Method	Feature	k=1	2	5	10
Cosine Classifier (baseline)	Standard	37.8	51.0	65.5	72.5
Cosine Classifier *Gen* (baseline)	Standard	42.6	53.9	66.4	72.6
Cosine Classifier *DMAS w/ in*	Standard	43.4	54.7	67.1	73.5
Cosine Classifier *DMAS w/ ex*	Standard	44.5	56.2	68.6	74.2
Cosine Classifier *DMAS w/ ex*↑	Standard	44.3	56.3	68.8	74.2
Cosine Classifier *DMAS w/ ex*↓	Standard	45.4	56.7	68.8	74.8
Cosine Classifier *DMAS (full)*	Standard	**46.5**	**58.3**	**69.7**	**75.1**

Table 3.2: Ablation studies (top-5 accuracy) on ImageNet1K **311-way** classification.

Comparisons with State of the Art. We compare our model with the state-of-the-art methods. We report 5-way 1-shot and 5-way 5-shot performance on three benchmarks: *mini*ImageNet, *tiered*ImageNet, and CUB, and 311-way k-shot on ImageNet1K. The results are summarized in Table 3.1. Under the same backbones, our model consistently achieves the best performance on all the datasets and across different sample-size regimes, even outperforming sophisticated methods, such as the attention based classifier 'Cosine Att. Weight' [76] and DeepEMD [273]. In particular, our 1-shot model outperforms state-of-the-art methods by significant margins, *e.g.*, 1.5% on *mini*ImageNet, 2% on *tiered*ImageNet, and 2.8% on CUB.

Ablation Analysis. To unpack the performance gain and understand the impact of different components, we perform a series of ablations on the challenging ImageNet1K dataset. Tables 3.2 summarizes the top-5 accuracies and the 95% confidence intervals for all number are of the order of 0.2%: : (1) **different pre-trained feature spaces** for hallucination – 'standard' (the feature backbone is a ResNet10 pre-trained using a standard cross-entropy linear classifier on base classes) vs. 'cosine' (the ResNet10 feature backbone is pre-trained using a cosine classifier); (2) **different types of classifiers** – prototypical net vs. cosine classifier; (3) **impact of different**

Method	k=1	5
ProtoNet [213]	50.01 ± 0.82	72.02 ± 0.67
MatchingNet [233]	51.65 ± 0.84	69.14 ± 0.72
Cosine Classifier [44]	44.17 ± 0.78	69.01 ± 0.74
Linear Classifier [44]	50.37 ± 0.79	73.30 ± 0.69
KNN [141]	50.84 ± 0.81	71.25 ± 0.69
DeepEMD [273]	54.24 ± 0.86	78.86 ± 0.65
DMAS (Ours)	**63.72 ± 0.29**	**81.24 ± 0.20**

Table 3.3: Cross-domain evaluation (*mini*ImageNet → CUB). Our model outperforms other baseline methods by large margins, showing the generalization of our learned hallucinator.

sources of supervision and progressive training. *w/ aug*: with standard data augmentation. *Gen*: with a plain hallucinator [244] trained using the classification loss on the query set solely. *DMAS w/ ex*: DMAS trained only under the guidance of the mentor *without progressive training*. *DMAS w/ ex*$\uparrow$: progressive guidance through *strengthening the mentor*. *DMAS w/ ex*$\downarrow$: progressive guidance through *weakening the student*. *DMAS w/ in*: DMAS trained only in a self-directed way through contrastive learning. *DMAS (full)*: trained under both (progressively) extrinsic and intrinsic supervision.

Robust to different types of pre-trained features and classifiers. Table 3.2 shows that DMAS can effectively hallucinate data in different types of pre-trained feature spaces and can work with different types of classifiers. Notably, DMAS achieves the best performance in a *homogeneous* setting, where the feature is pre-trained by using a cosine classifier and the final classification model is also a cosine classifier.

Extrinsic guidance from mentor. From Table 3.2, we can observe that DMAS significantly outperforms baselines by benefiting from the extrinsic guidance of the mentor. There are 5.8% improvement when combining with the prototypical network and 6.7% improvement when combining with the cosine classifier. More importantly, DMAS outperforms the plain hallucinator [244] which is trained using the classification loss only. Note that both the baselines and DMAS use the same amount of data for meta-training on base classes.

Intrinsic supervision. Table 3.2 also shows that DMAS trained only with the intrinsic supervision already outperforms the baselines. The improvement is more pronounced when there are very few examples, *e.g.*, 5.6% improvement when $k = 1$. This implies the importance of preserving the label consistency between hallucinated and real examples. In addition, the full DMAS model achieves the best performance, demonstrating that *the extrinsic supervision and the intrinsic supervision are complementary to each other*.

Strengthening the mentor vs. weakening the student. We compare two directions for progres-

Method	Backbone	k=1	5
MetaOptNet [126]	ResNet12	62.64 ± 0.61	78.63 ± 0.46
MetaOptNet + *Gen* [244]	ResNet12	63.46 ± 0.43	80.02 ± 0.28
MetaOptNet + DMAS (Ours)	ResNet12	**64.55 ± 0.64**	**80.42 ± 0.46**
S2M2 [165]	WRN-28-10	63.90 ± 0.18	81.03 ± 0.11
S2M2 + *Gen* [244]	WRN-28-10	63.37 ± 0.56	81.23 ± 0.19
S2M2 + DMAS (Ours)	WRN-28-10	**65.35 ± 0.63**	**83.55 ± 0.41**
DeepEMD [273]	ResNet12	65.91 ± 0.82	82.41 ± 0.56
DeepEMD + *Gen* [244]	ResNet12	64.73 ± 0.30	79.92 ± 0.21
DeepEMD + DMAS (Ours)	ResNet12	**67.42 ± 0.28**	**83.74 ± 0.20**

Table 3.4: Ablation study on the generalizability of our approach and additional comparisons with state of the art on *mini*ImageNet. Our DMAS hallucinator is **general and can work with different types of classification models and different backbone models** to *consistently* improve their performance. In addition, DMAS consistently outperforms the plain hallucinator [244].

sive guidance by strengthening the mentor (*w/ ex*↑) and weakening the student (*w/ ex*↓). We use a logarithmic scale when changing the number of examples on which the student or mentor model is trained [217, 243]. As shown in Table 3.2, both directions outperform the normal guidance without progression (*w/ ex*), and weakening the student achieves better results. It comes from the fact that, if both mentor and student start being weak, the learning problem could actually be hard due to the high variance of both mentor and student.

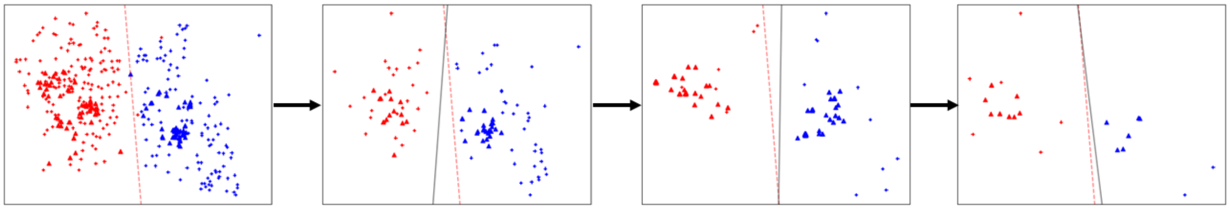


Figure 3.4: Visualization with t-SNE of the evolution of the decision boundary for two *novel* classes, when meta-training our DMAS hallucinator through progressive guidance by weakening the student. **Best viewed in color with zoom.**

Comparisons with standard data augmentation. Table 3.2 shows that our learned data hallucination outperforms meta-learning with standard hand-crafted data augmentation ('w/ aug'), which includes random crop, random horizontal flip, and color jittering as in [44], indicating the importance of exploiting shared intra-class variation.

Cross-Domain Evaluation. So far, we have focused on the within-domain scenario. Now we consider the cross-domain scenario, which allows us to investigate the generalization of our

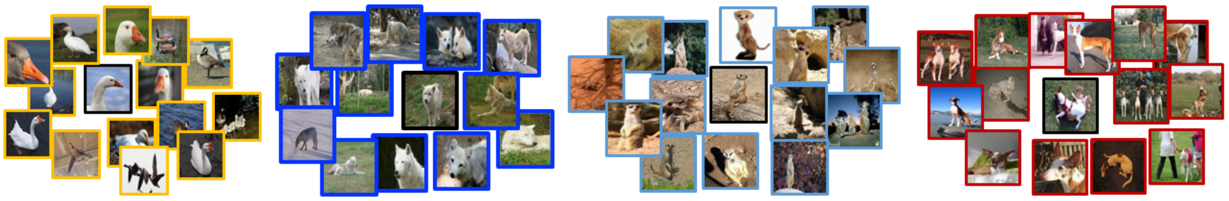


Figure 3.5: Visualization of nearest neighbor real images of hallucinated examples for four *novel* classes. **Best viewed in color with zoom.**

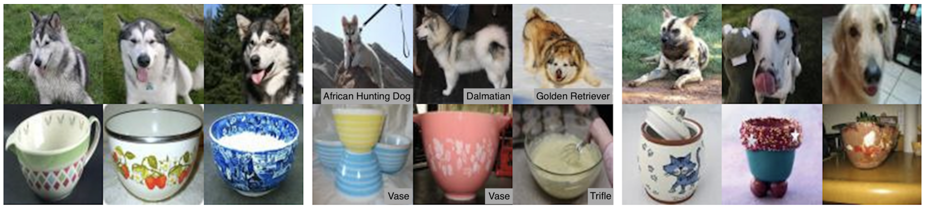


Figure 3.6: Visualization of classification results of two *novel* classes (Top row: malamute; bottom row: mixing bowl) and comparison between our DMAS hallucinator and the plain hallucinator [244].

DMAS hallucinator and understand the effects of domain shifts. Following the cross-domain setup in [44, 273], the experiment in Table 3.3 shows that our DMAS hallucinator trained on *mini*ImageNet is effective for never-before-seen classes on CUB *without any fine-tuning*.

DMAS as a General Plug-and-Play Module. Table 3.4 further shows the generalizability of our approach – the DMAS hallucinator can work with different types of classification models and different backbone models to consistently improve their performance. To fully investigate the impact of DMAS and for a fair comparison, we conduct experiments on *mini*ImageNet with the same training setups (*e.g.*, backbones, data augmentation techniques, and training strategies) as the state-of-the-art approaches [126, 165, 273]. In all cases, DMAS can be seamlessly incorporated into these approaches (denoted as '$+$'), and substantially improve their performance, *e.g.*, 1.9% improvement when combining with MetaOptNet [126] and 1.5% improvement when combining with S2M2 [165] under the challenging 1-shot setting.

Comparisons with the plain hallucinator. Table 3.4 also shows that DMAS *consistently* outperforms the plain hallucinator [244] for different types of models (Table 3.2 has already shown this for ProtoNet and cosine classifier). More importantly, 'DeepEMD + [244]' is *worse* than the plain DeepEMD; a similar phenomenon is observed with S2M2 in the 1-shot case. These results suggest that, *while DMAS is general, [244] is not a general module for different few-shot models*. For more sophisticated models (S2M2 and DeepEMD), solely using the classification loss as in [244] is insufficient to adjust the hallucinator to produce effective samples. This further verifies the importance of extrinsic and intrinsic supervision.

Visualizations. To further understand how our model helps learning a classifier and refining the hallucinator, we conduct visualizations on ImageNet1K. We first visualize in Figure 3.4 the evolution of the decision boundary for two *novel* classes during progressive guidance by weakening the student using t-SNE [230]. Real examples (small dots) are progressively removed, and hallucinated examples (triangles) are generated in a way that helps maintain the student decision boundary (black solid line) as close as possible to the desired decision boundary that would be formulated by a large set of real examples (red dashed line). We observe that PCA visualization has a similar phenomenon. We then visualize in Figure 3.5 the hallucinated examples in the pixel space, using their nearest neighbor real images in the feature space. For each class, the single black framed image comes from the original dataset and is used as a seed for generating new examples. Color framed images correspond to the nearest neighbor real images of the hallucinated examples in the feature space. Finally in Figure 3.6, we compare our approach with the state-of-the-art meta-learned hallucinator [244] and show that ours is able to recognize a large range of visual variations. The left block shows images correctly classified by both approaches. The middle block shows images that are misclassified by [244] as other classes (with predicted class names overlaid on the images), but correctly classified by our approach. The right block shows images from other classes that are misclassified by [244] as the target class, but correctly classified by our approach. In these examples, our classifier is able to recognize objects with different poses and view points, whereas [244] fails to distinguish between similar classes.

3.6 Summary

We present an approach to few-shot classification that uses a dual mentor- and self-directed hallucinator to generate additional examples. This is achieved by exploiting two important natural supervision signals that facilitate data hallucination in a way that most improves the classification performance, and is trained end-to-end through meta-learning. Our hallucinator can be inserted as a plug-and-play module into different classification models. The extensive experiments demonstrate our state-of-the-art performance on the widely-benchmarked few-shot datasets in various scenarios.

Part II

Weak Supervision for Multimodal Representation Learning

Chapter 4

Learning Generalizable Representations from Image-Text Pairs

This chapter investigates the use of image-text pairs as weakly supervised signals to learn generalization and effective multimodal representations. By aligning visual and textual modalities, the proposed approach mitigates the need for extensive human annotations and fosters the development of robust representations that transfer effectively across tasks and domains.

4.1 Overview

In previous chapter, we show that external (*i.e.*, implicit and explicit) knowledge is required in multimodal tasks. While incorporating external knowledge is complementary to open-domain multimodal understanding, it still needs lots of human efforts to construct (*e.g.*, the construction of of Wikidata). In this chapter, we focus on the rsearch question: can we learn multimodal representations without expensive human efforts (*e.g.*, human annotations) in a single, unified architecture?

We show that vision-language transformers can be learned without human labels (e.g. class labels, bounding boxes, etc). Existing work, whether explicitly utilizing bounding boxes [49, 159, 223] or patches [115], assumes that the visual backbone must first be trained on ImageNet [198] class prediction before being integrated into a multimodal linguistic pipeline. We show that this is not necessary and introduce a new model **V**ision-**L**anguage from **C**aptions (**VLC**) built on top of Masked Auto-Encoders [91] that does not require this supervision. In fact, in a head-to-head comparison between ViLT, the current state-of-the-art patch-based vision-language transformer which is pretrained with supervised object classification, and our model, **VLC**, we find that our approach 1. outperforms ViLT on standard benchmarks, 2. provides more interpretable and intuitive patch visualizations, and 3. is competitive with many larger models that utilize ROIs trained on annotated bounding-boxes.

4.2 Motivation

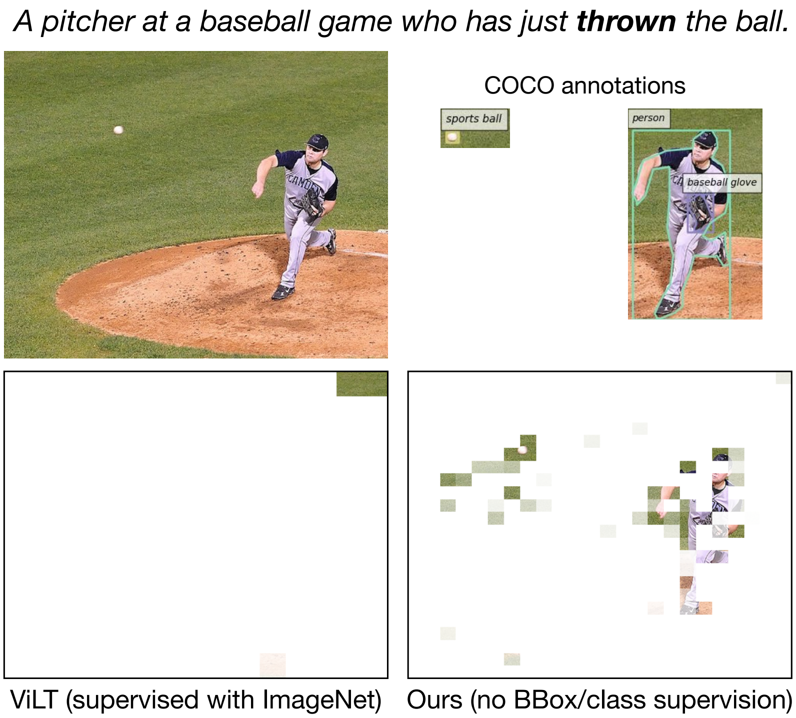

A pitcher at a baseball game who has just **thrown** the ball.
COCO annotations
sports ball
person
baseball glove
ViLT (supervised with ImageNet)
Ours (no BBox/class supervision)

Figure 4.1: We present an image with its corresponding annotations and caption. Visualized are the model's top aligned patches with the word **thrown**. Note, ViLT often chooses a single (predictive) patch, where our model **VLC** produces a more meaningful (if diffuse) distribution over the relevant patches.

Should vision guide language understanding or does language structure visual representations? Vision-language (VL) transformers have put language first. Most popular vision-language transformers [49, 139, 159, 223] only integrate vision from selected bounding boxes extracted by pretrained ImageNet [198] classifiers. In this paradigm, the bag of visual tokens are embedded into an existing linguistic space (*i.e.*, the lexical embeddings of BERT [62]).

The introduction of ViT [66] empowered the community to flip the paradigm. Notably, ViLT [115] initializes with ViT [66], so the initial semantic representation is vision based and language projects into the patch space. This flipped paradigm places visual representations as the initial conceptual space to which language must adhere. Additionally, there are engineering benefits to this paradigm as it removes the computationally expensive need for Region of Interest (ROI) extraction. However, because ViT is trained with supervised class labels, its representation may be constrained by the limited concepts ImageNet covers, the space is still somewhat linguistic in nature when initialized, and requires expensive data annotation, a hindrance to scaling to arbitrarily many visual classification categories.

We take the important next step and remove the need for supervised pretraining. An unsupervised visual semantics is learned via Masked Auto-Encoders [91] before language is integrated. This leads to both a better performing and more general model. In addition, every component can be improved and scaled with unsupervised and weakly aligned data – removing the need for future annotation efforts while still scaling to open-vocabulary domains in the wild.

Our **V**ision-**L**anguage from **C**aptions (**VLC**) model matches or outperforms nearly all vision-language transformers despite being 1. Smaller, 2. Avoiding use of ROIs, and 3. Not leveraging object-level supervised labels. We evaluate across several popular benchmarks in addition to retrieval and probing. Performance also continues to improve with data and model size scaling, and as it relies only on weak alignment of image-text pairs, future work with access to large compute may be able to continue driving up performance. Ablation study shows masked modeling on images can consistently improve the performance on downstream tasks which is in sharp contrast to existing approaches. Finally, we provide several analyses on the underlying patch/lexical representations to understand what our models are learning and guide future VL transformer research.

4.3 Prior Work

Vision-Language Modeling. Based on how they encode images, most existing works on vision-language modeling fall into three categories. The first category [49, 139, 140, 143, 159, 187, 214, 223, 276] focuses on using pre-trained object detectors to extract region-level visual features (e.g., by Faster R-CNN [196]). In particular, OSCAR [143] and VinVL [276] further boost the performance by feeding additional image tags into the transformer model. However, extracting region-level features requires pretrained object detectors with high-resolution inputs that can be time-consuming. To tackle these two issues, the second category [97, 98, 103] proposes to encode images by using grid features from convolutional neural networks. SOHO [98] discretizes the grid features by a learnable vision dictionary, and feeds the discretized features to their cross-modal module. The third category [67, 133, 189, 263] uses a Vision Transformer (ViT) [66] as the image encoder and designs different objective functions for vision-language pretraining. The most similar to our work is ViLT [115]. ViLT does not use pretrained object detectors or extra visual embedders for visual embedding, but still needs weights pretrained on ImageNet-21K for initialization. Different from the previous work, we show such supervised initialization, pretrained object detectors or visual embedders are not necessary. While momentum distillation [133] and image-text contrastive loss [133, 134] are shown effective in previous work, such techniques are orthogonal to our work and not included in our discussion.

Masked Language Modeling. Masked language modeling (MLM) and its auto-regressive counterparts are widely used in natural language processing for learning text representations. MLM [62] trains a model to predict a random sample of input tokens that have been masked in a multi-class setting. In vision-language pretraining, MLM has shown useful to enforce the consistency across modalities [67, 115, 143, 276]. In vision-language modeling, we randomly mask some of the input tokens, and the model is trained to reconstruct the original tokens given the masked tokens and their corresponding visual inputs. To be consistent with previous work, we follow the default settings of training BERT [62] for masked language modeling.

Masked Image Modeling. Masked image modeling (MIM) is a pretext task to learn representations from images corrupted by masking. Inspired by the success of masked language modeling (MLM) in NLP, different masked prediction objectives have been proposed for image tasks. iGPT [42] predicts unknown pixels of a sequence. ViT [66] predicts mean colors of masked patches. BEiT [19] proposes to use a pre-trained discrete variational autoencoder (dVAE) [192] to encode masked patches. MaskFeat [246] predicts HoG [58] features of the masked image regions. SimMIM [258] and MAE [91] predict RGB values of raw pixels by direct regression. MIM has also been explored in the field of vision-language representation learning by either regressing the masked feature values [49, 67, 115, 223] or predicting a distribution over semantic classes for the corresponding image region [49, 159, 214]. In contrast to previous approaches [49, 67, 115] that show MIM does not contribute to or hurt the performance on downstream tasks, we show that using MIM can consistently improve the performance as the training steps increase.

4.4 Vision-Language from Captions

Model Architecture

Our aim is a vision-language transformer that can be trained without the need for expensive object-level supervised labels (*e.g.*, class labels or object bounding boxes). More concretely, our results empirically show that such object based supervised signals are *not* necessary for vision-language pretraining. To this end, we use a ViT-based framework to learn multi-modal representations by 1) intra-modal reconstruction through masked image/language modeling; 2) inter-modal alignment through image-text matching. The architecture of our proposed **VLC** framework is illustrated in Figure 5.2. **VLC** consists of a modality-specific projection module 4.4,

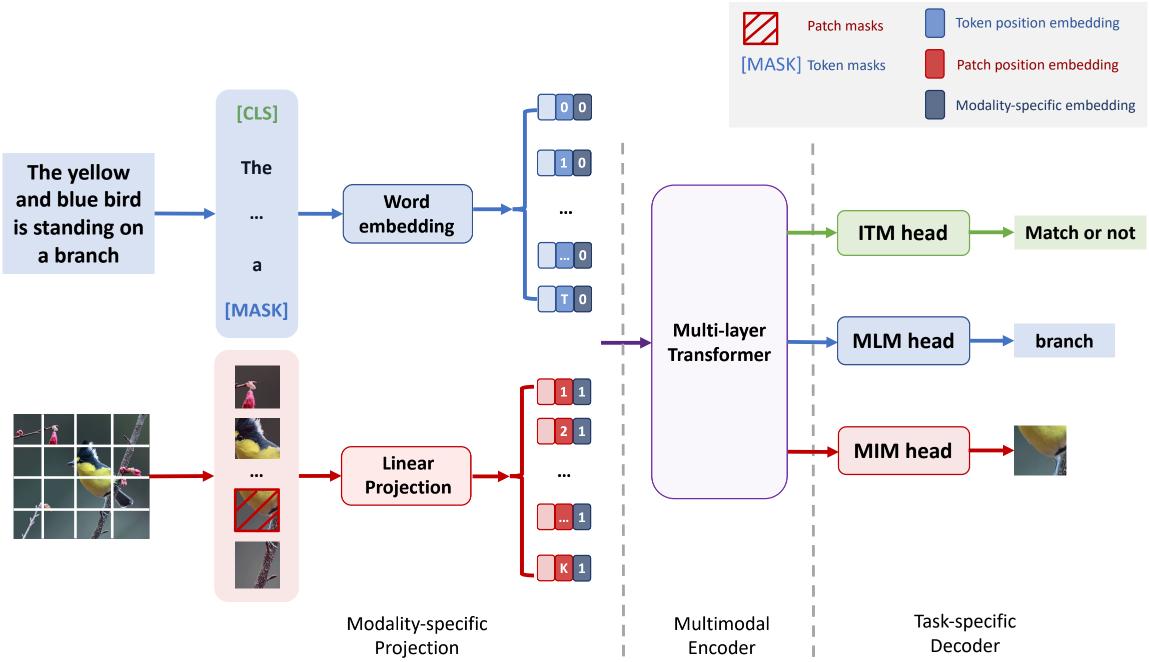


Figure 4.2: The overall architecture of our **VLC** model. Our model consists of three modules: (1) Modality-specific projection. We use a simple linear projection to embed patched images and a word embedding layer to embed tokenized text; (2) Multi-modal encoder. We use a 12-layer ViT [66] initialized from MAE [91] (ImageNet-1K without labels) as our backbone; (3) Task-specific decoder. We learn our multi-modal representations by masked image/language modeling and image-text matching which are only used during pre-training. We use a 2-layer MLP to fine-tune our multi-modal encoder for downstream tasks. Importantly, we find that the masked image modeling objective is important throughout second-stage pre-training, not only for initialization of the visual transformer.

a multi-modal encoder 5.4 and three task-specific decoders 4.4. We aim for minimal visual and textual embedding designs during pretraining.

Modality-specific Projection Module

While most of existing methods rely on complex ResNeXt [97] or object detection components [49, 143, 159, 276], we use a trainable *linear projection* layer to map flattened visual patches to the visual embedding space. The patch embeddings are represented as $\mathbf{v} = \{v_1, ..., v_K\} \in \mathbb{R}^{K \times d}$, where K is the number of image patches and d is the hidden dimension of our model. For text embedder, we follow BERT [62] to tokenize the input sentence into WordPieces [251]. We then adopt a word embedding lookup layer to project tokenized words to the textual embedding space.

Here we use $\mathbf{w} = \{w_{CLS}, w_1, ..., w_T\} \in \mathbb{R}^{T \times d}$ to represent the token embeddings, where T is the number of tokens and the special token `CLS` denotes the start of the token sequence. We encode patch and token positions separately by $v^{pos} \in \mathbb{R}^{1 \times d}$ and $w^{pos} \in \mathbb{R}^{1 \times d}$. We use $v^{type} \in \mathbb{R}^{1 \times d}$ and $w^{type} \in \mathbb{R}^{1 \times d}$ as modality-type embeddings to distinguish the modality difference between patch and token embeddings. The final representations of each patch v_i and token w_j are calculated as

$$\hat{v}_i = \text{LayerNorm}(v_i + v_i^{pos} + v^{type}), \quad \text{and} \tag{4.1}$$

$$\hat{w}_j = \text{LayerNorm}(w_j + w_j^{pos} + w^{type}). \tag{4.2}$$

Multi-modal Encoder

To learn the contextual representations from both visual and textual modality, we follow single-stream approaches [49, 115] and use the ViT-B/16 architecture as our multi-modal encoder. ViT-B/16 consists 12 alternating layers of multiheaded self-attention (MSA) and MLP blocks. LayerNorm comes before every block and residual connections after after every block [66].

We use a merged-attention [67] mechanism to fuse the visual and textual modalities. More specifically, we concatenate the token and patch embeddings together as $\{\hat{w}_{CLS}, \hat{w}_1, ..., \hat{w}_T, \hat{v}_1, ..., \hat{v}_K\}$, then feed them into the transformer. We use the hidden states h at the output of the last layer of the encoder as the contextual representations $\{h_{CLS}, h_1^w, ..., h_T^w, h_1^v, ..., h_K^v\}$. In sharp contrast to existing approaches that use object detectors, visual detectors pretrained with supervised labels or pretrained language models (*e.g.*, BERT, Roberta), we initialize our model with MAE pretrained on ImageNet-1K with no labels.

Pretraining Objectives

To learn a universal visual and textual representation for vision-and-language tasks, we apply self-supervised methods to pre-train a model on a large aggregated dataset. Unlike previous approaches that only mask text tokens, we randomly mask both image patches and text tokens simultaneously. We train our model with three objectives: masked image modeling (MIM), masked language modeling (MLM) and image-text matching (ITM).

Masked Language Modeling. In language pretraining, MLM randomly masks input tokens, and the model is trained to reconstruct the original tokens based on unmasked context. Following BERT [62], we randomly mask text tokens with a probability of 0.15, and replace the masked ones $\mathbf{w_m}$ with a special token `[MASK]`. The goal is to predict the masked tokens based on both non-masked text tokens $\mathbf{w}_{\backslash \mathbf{m}}$ and image patches $\mathbf{v}_{\backslash \mathbf{m}}$. The learning target $\mathcal{L}_{MLM}$ can be formulated as

$$\mathcal{L}_{MLM} = -\mathbb{E}_{(\mathbf{w},\mathbf{v})\sim D} \log p(\mathbf{w_m} | \mathbf{w}_{\backslash \mathbf{m}}, \mathbf{v}_{\backslash \mathbf{m}}). \tag{4.3}$$

We use a linear layer with default parameters [62] as the MLM head to output logits over the vocabulary, which are used to compute the negative log likelihood loss for the masked text tokens.

Masked Image Modeling. Existing approaches explore MIM either by regressing the masked features values [49, 115, 263] or by predicting a distribution over semantic classes for a certain image region [49, 67, 159]. In contrast, we follow MAE [91] to randomly mask image patches with a probability of 0.6, and reconstruct the missing pixels based on both non-masked tokens $\mathbf{w}_{\backslash\mathbf{m}}$ and patches $\mathbf{v}_{\backslash\mathbf{m}}$. The learning target $\mathcal{L}_{MIM}$ can be formulated as

$$\mathcal{L}_{MIM} = \mathbb{E}_{(\mathbf{w},\mathbf{v})\sim D} f(\mathbf{v}_{\mathbf{m}}|\mathbf{w}_{\backslash\mathbf{m}}, \mathbf{v}_{\backslash\mathbf{m}}), \tag{4.4}$$

where the feature regression objective f is to regress the masked image patch representations to pixel values. We use 8-layer transformer as the MIM head r. For a masked image patch v_i, the objective f can be formulated as: $f(v_i|\mathbf{w}_{\backslash\mathbf{m}}, \mathbf{v}_{\backslash\mathbf{m}}) = ||r(h_i^v) - v_i||^2$. Each output of the MIM head is a vector of pixel values representing a patch. Different from the observations in ViLT [115] and METER [67], we show MIM can consistently improve the performance on downstream tasks as training steps increase.

Image-Text Matching. Given a batch of image and text pairs, the ITM head identifies if the sampled pair is aligned. We randomly replace the aligned image with a different one with a probability of 0.5. We use the special token `[CLS]` as the fused representation of both modalities, and feed h_{CLS} to the ITM head. The learning target $\mathcal{L}_{ITM}$ can be formulated as

$$\mathcal{L}_{ITM} = -\mathbb{E}_{(\mathbf{w},\mathbf{v})\sim D} \log p(y|\mathbf{w}, \mathbf{v}), \tag{4.5}$$

Where $y \in \{0, 1\}$ indicates whether the image and text are matched ($y = 1$) or not ($y = 0$). We use a single linear layer as the ITM head and compute negative log likelihood loss as our ITM loss. We weight the pretraining objectives equally so the full pre-training objective is:

$$\mathcal{L} = \mathcal{L}_{MLM} + \mathcal{L}_{ITM} + \mathcal{L}_{MIM} \tag{4.6}$$

For a fair comparison with existing approaches, We do not include image-text contrastive loss [133, 134], momentum distillation [133] and other techniques in our implementation.

4.5 Empirical Evaluation

We conduct extensive experiments on a diversified set of vision-language benchmarks, including image-text retrieval, visual question answering and natural language for visual reasoning. We evaluate our pretrained model to each downstream task through end-to-end fine-tuning. To further show the generalization ability of our pre-trained model, we examine our model on ImageNet-1K classification task following common practice [66, 91]. We also evaluate our model on the open-domain VQA task that requires commonsense reasoning of the scene depicted in the image.

		Text Retrieval						Image Retrieval					
Model		Flickr30K (1K)			MSCOCO (5K)			Flickr30K (1K)			MSCOCO (5K)		
	Params	@1	@5	@10	@1	@5	@10	@1	@5	@10	@1	@5	@10
ALBEF† [133]	163M	94.3	99.4	99.8	73.1	91.4	96.0	**82.8**	**96.7**	**98.4**	56.8	81.5	89.2
VinVL$_{\text{LARGE}}$ [276]	452M	-	-	-	75.4	92.9	96.2	-	-	-	**58.8**	83.5	90.3
UNITER$_{\text{LARGE}}$ [49]	371M	87.3	98.0	99.2	65.7	88.6	93.8	75.6	94.1	96.8	52.9	79.9	88.0
METER-Swin$_{\text{BASE}}$ [143]	288M	92.4	99.0	99.5	76.2	93.2	96.8	79.0	95.6	98.0	54.9	81.4	89.3
PixelBERT [97]	144M	87.0	98.9	99.5	63.6	87.5	93.6	71.5	92.1	95.8	50.1	77.6	86.2
ViLT [115]	86M	83.5	96.7	98.6	61.5	86.3	92.7	64.4	88.7	93.8	42.7	72.9	83.1
VLC-Base (ours – 5.6M)	86M	89.2	99.2	99.8	71.3	91.2	95.8	72.4	93.4	96.5	50.7	78.9	88.0
VLC-Large (ours – 5.6M)	307M	**94.4**	**99.6**	**99.9**	**76.7**	**94.5**	**97.3**	79.1	95.8	98.2	58.4	**84.0**	**91.1**

Table 4.1: We compare our model with several state of the art bounding box based and supervised methods. We see substantial gains across all settings. †ALBEF uses specifically designed coarse-to-fine objectives for the image-text retrieval task.

Pre-training Datasets

Following previous work [49, 67, 115, 133], our pre-training corpus comprises four commonly used vision-language datasets including COCO [151], Visual Genome [120], Google Conceptual Captions [208] and SBU Captions [178], totalling 4.0M unique images and 5.1M image-text pairs. To show the benefits of data-scaling, we also use the VinVL [276] pretraining data which includes Flickr30k [268], GQA [99], VQA [80], VG-QAs [120] and a subset of OpenImages [119]. This larger pre-training corpus contains 5.65M unique images (see detailed statistics in Appendix A.1). Future work can trivially grow the size of the corpus by including large-scale web crawls.

Downstream Tasks

We evaluate our model on image-text retrieval tasks including Flickr30K [185] and MSCOCO [151], and image-text understanding tasks including VQAv2 [80] and NLVR^2 [216]. For retrieval tasks, we follow the standard splits and evaluate our models in the finetuning settings. For VQAv2, we follow the standard practice [49, 133] to train the models with both training, validation and additional question-answer pairs from Visual Genome while reserving $1,000$ validation samples for internal validation. For ablation and analysis, we mainly focus on VQAv2. More evaluation details can be found in Appendix A.3.

Implementation Details

We pretrain two variants of the multi-modal encoder which uses a 86M parameter ViT-B/16 denoted as **VLC**-Base and 307M parameter ViT-L/16 denoted as **VLC**-Large. Both variants are initialized with MAE pre-trained on ImageNet-1K without labels. For text inputs, we tokenize text with the *bert-base-uncased* and *bert-large-uncased* tokenizer, respectively. The text embedding parameters are learned from scratch, in lieu of loading pre-trained BERT weights. We randomly mask image patches with a probability of 0.6 and text tokens with a probability 0.15. To accelerate training, we follow MAE [91] and skip the mask token `[MASK]` in the encoder and only apply it in the lightweight decoder. We use AdamW [158] with a weight decay of 0.01. The learning rate is warmed-up to $1e^{-4}$ in the first 10% of total training steps and is decayed to zero for the rest of the training following a linear schedule. During pre-training, we resize the shorter edge of input images to 384, take random image crops of resolution 384×384, and apply RandAugment [56]. We pre-train for $200k$ steps with a batch size of $4,096$. For the parameter estimation, we exclude the textual embedder as it is shared by all vision-language transformers. We also exclude the parameters of all the auxiliary heads as they are only required during pretraining. Unless otherwise specified, we use the *base* version of **VLC** for visualizations and ablation studies.

For all downstream tasks, we fine-tune our model with a learning rate of $5e^{-4}$ for 10 epochs. We use a layer-wise learning rate decay [55] of 0.5. We use 576×576 as the input image resolution for the VQA task and 384×384 for NLVR^2 and image-text retrieval tasks.

Visual Question Answering (VQA [80]). Given an input image and a question, the VQA task is to predict an answer from the visual content. We conduct experiments on VQAv2 dataset [80] that is built on MSCOCO. It contains 83K images for training, 41K for validation, and 81K for testing. Following previous work [49, 133, 223], we use the training, validation splits and additional question-answer pairs from Visual Genome while reserving $1,000$ validation image-question pairs for internal validation. We report performance on the test-dev and test-std splits. We use a 2-layer MLP with a hidden size of $1,536$ to adapt **VLC** to the VQA task. We follow the standard practice [115] to convert the task to a multilabel classification task with $3,192$ answer classes.

Natural Language for Visual Reasoning (NLVR^2 [216]). Given a triplet of two images and a description, this task is to predict whether this description describes a pair of images. Following previous work [49, 115], we use the *pair* method which treats one input sample as two image-text pairs by repeating the text twice. Each pair is passed through our model and we take the concatenation of two pooled representation `[CLS]` from our model as the representation of one input sample. Similar to the settings of the VQA task, we use a 2-layer MLP with a hidden size of $1,536$ to adapt **VLC** to the NLVR^2 task.

Image-Text Retrieval. Image-Text retrieval contains two subtasks: image-to-text retrieval (TR) and text-to-image retrieval (IR). We evaluate our pre-trained models on the Karpathy splits [110] of MSCOCO [151] and Flickr30K [185] in fine-tuning settings. MSCOCO contains 123K images, and each image has five corresponding human-written captions. We split the data into 82K/5K/5K training/validation/test images. To be consistent with previous work [49, 115], we use the additional 30K images from MSCOCO validation set to improve the performance. Flickr30K contains 31K images with five captions for each image. We split the data into 30K/1K/1K as the training/validation/test set.

Adapt VLC to Downstream Tasks

Image-Text Retrieval Tasks. We begin with a proof of concept experiment, evaluating our model on the Karpathy splits of the Flickr30K [185] and MSCOCO [151] benchmarks. Table 4.1 compares VLC to strong multimodal transformers which leverage ROIs, more parameters, and are pretrained on ImageNet classification. Note that as most of detection-based models have the advantage of using Faster R-CNN [196] pre-trained on VG [120] or MSCOCO [151].

The closest comparison to **VLC**-Base is ViLT as it is the same model size, though still requires more supervised data in the form of ImageNet classification pretraining for ViT [66][1]. When comparing to dual-encoder models, our **VLC**-Large achieves competitive results across all settings. ALBEF uses pre-trained ViT and BERT model for initialization. Additionally, it specifically designs the coarse-to-fine objectives while we directly fine-tune the pre-trained ITM head for retrieval tasks. Thus we treat ALBEF as a strongest available baseline.

Image-Text Understanding Tasks. Table 4.2 presents **VLC** results on two popular image-text understanding datasets: VQAv2 and NLVR^2. We use the same training data as ViLT denoted as 4M and VinVL denoted as 5.6M.

Comparison to models supervised/initialized with ImageNet bounded boxes. Most of these models use object detectors pretrained on VG [120] or MSCOCO [151] to extract region features. Object detectors help in VQA tasks as they mainly ask about objects. Within the similar scale of pretraining data, our model achieves competitive performance on both tasks. Note that our model uses 384×384 or 576×576 as input resolution during our fine-tuning stages. This resolution is much lower compared with previsou work using 800×1333 [49, 159]. In particular, *VinVL* [276] has a multi-stage pre-training for its object detector that has access to ImageNet-5K [257] (6.8M images from 5K classes) and four object detection datasets [119, 120, 151, 207] (2.5M images with bounding box annotations). In addition, *VinVL* uses pretrained BERT as the multimodal

[1] ViLT uses ViT-B/32 pretrained with ImageNet-21K and finetuned on ImageNet-1K with supervised labels.

Model		VQAv2		NLVR^2	
	Params	test-dev	test-std	dev	test
Supervised ImageNet Bounded Boxes					
ViLBERT [159]	274M	70.55	70.92	-	-
LXMERT [223]	240M	72.42	72.54	74.90	74.50
VisualBERT [139]	170M	70.80	71.00	67.4	67.0
UNITER$_{\text{LARGE}}$ [49]	371M	73.82	74.02	79.12	79.98
OSCAR$_{\text{LARGE}}$ [143]	371M	73.61	73.82	79.12	80.37
VinVL$_{\text{LARGE}}$ † [276] (5.6M)	452M	76.52	76.60	**82.67**	**83.98**
Supervised ImageNet Classes					
METER-Swin$_{\text{BASE}}$ ‡ [68]	288M	76.43	76.42	82.23	82.47
ALBEF [133]	163M	74.54	74.70	80.24	80.50
Visual Parsing [263]	180M	74.00	74.17	77.61	78.05
PixelBERT [97]	144M	74.45	74.55	76.5	77.2
ViLT [115]	86M	71.26	-	75.70	76.13
No supervised classes or bounding boxes					
VLC-Base (ours – 4M)	86M	72.98	73.03	77.04	78.51
VLC-Base (ours – 5.6M)	86M	74.02	74.0	77.70	79.04
VLC-Large (ours – 5.6M)	307M	**76.95**	**77.02**	82.27	83.52
Pre-trained or initialized with $> 10M$ data					
METER-CLIP-ViT$_{\text{BASE}}$ [68] (4M)	280M	77.68	77.64	82.33	83.05
X-VLM [272] (16M)	216M	78.22	78.37	84.41	84.76
BLIP [134] (129M)	252M	78.25	78.32	82.15	82.24
OFA [239] (54M)	930M	82.0	82.0	-	-
CoCa [269] (4.8B)	2.1B	82.3	82.3	86.1	87.0

Table 4.2: We compare our model with state-of-the-art pre-trained methods on vision-language understanding tasks. Our model (**VLC**), unlike all others, is only pre-trained with weakly-aligned image-caption pairs. Again, our approach matches or outperforms larger and more hevaily supervised approaches within a similar scale of training data and model size. †VinVL uses the object detector trained with $6.8M$ labeled imageNet images and 2.5M images with bounding box annotations. ‡METER-Swin$_{\text{BASE}}$ uses Swin-B trained with $14M$ labeled ImageNet as the image encoder and pretrained Roberta as the text encoder.

encoder. Our **VLC**-Large, which has a similar model size, achieves better performance on the

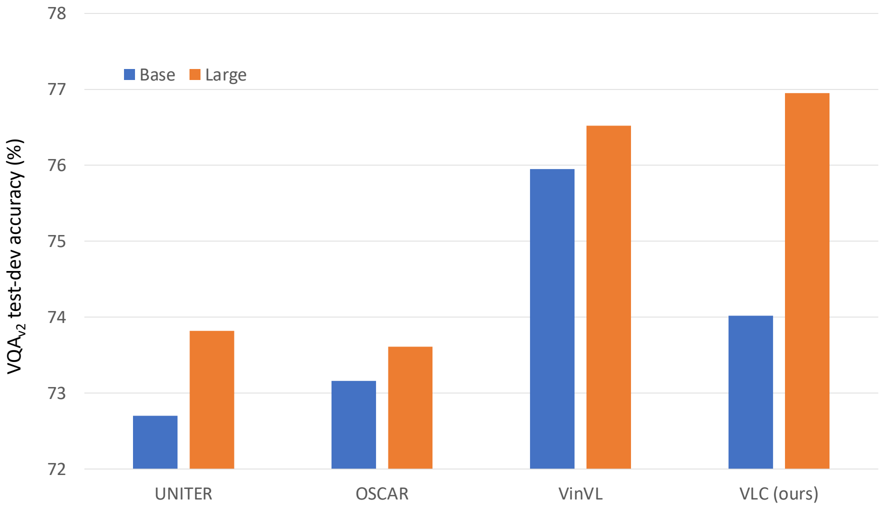


Figure 4.3: Comparison with our model with state-of-the-art pre-trained methods using different model size. While models initialized with supervised data provide strong priors, our **VLC** approach has the most substantial improvement when scaling the model size.

VQA task and competitive results on the NLVR task without any supervised initialization.

Comparison to models with supervised ImageNet classes. Most of these approaches use additional visual embedders together with a pretrained BERT as their backbones. For example, ALBEF [133], Visual Parsing [263], PixelBERT [97] use pre-trained ViT-B/16, Swin transformer, ResNeXt-152 as their visual embedder, respectively. All these embedders are trained with labeled ImageNet data. In particular, *METER-Swin$_{BASE}$* uses Swin-B/16 pretrained with more than 14M labeled ImageNet22K images as the image encoder and pretrained Roberta as the text encoder. Experiments show that our model achieves better results than larger and more heavily supervised approaches.

Note that there are some other baselines which have access to a much larger scale of data during pretraining. For example, METER-CLIP-ViT$_{\text{BASE}}$ uses CLIP as the image encoder which is trained with 400M image-text pairs. X-VLM [272] uses the same image encoder as METER-Swin$_{\text{BASE}}$ but with extra bounding box annotations and two times larger training data. While these models achieve higher performance, the comparisons are out of the scope.

Ablation Study

To understand the impact of different components, we ablate and compare variants of our model (*i.e.*, pretraining objectives and mode sizes) and report VQAv2 test-dev accuracy. In Figure 4.3, we compare our **VLC** with three baselines that use pretrained object detectors and BERT for initialization. We see that while supervised initialization provides strong priors for the VQA task

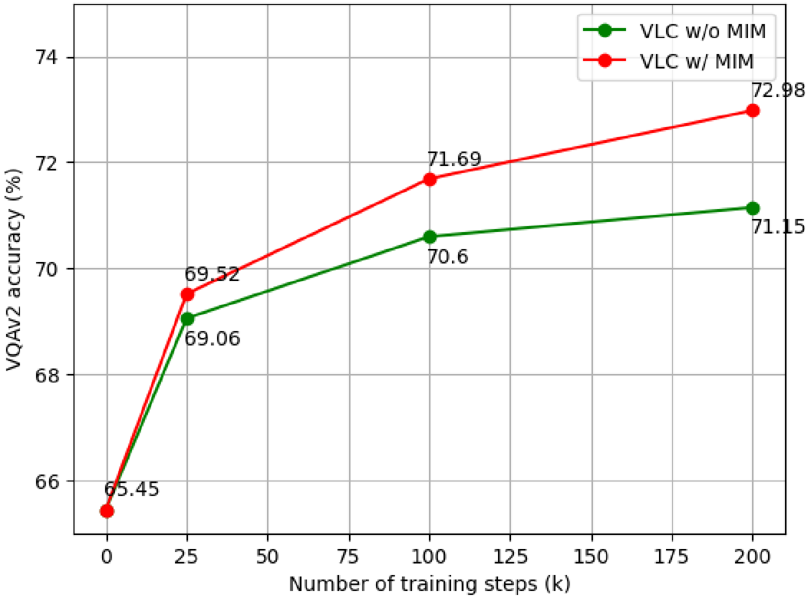


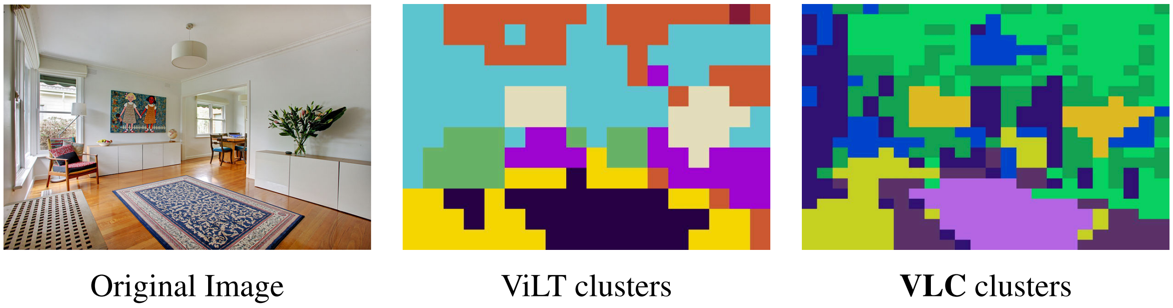
Figure 4.4: Ablation study on masked image modeling (MIM). Our experiments show that **VLC**-Base with MIM consistently outperforms the variant without MIM. Further increase in training steps enlarges the improvement on the VQAv2 task.

(*e.g.*, VinVL), scaling the model size only has a marginal improvement. As a comparison, our model is initialized with MAE pretrained on ImageNet1K without labels. There is a substantial gain when scaling to a larger model.

In Figure 4.4, we conduct an ablation study on masked image modeling (MIM). We train **VLC**-Base with 4M image-text pairs. As the training steps increase, there is a consistent improvement for **VLC** with MIM. This contrasts to findings in previous work [67, 115, 133].

4.6 Understanding the Models

While simpler and more efficient, patch-based models differ in important ways from traditional bounding-box based approaches. In particular, while the visual stack is traditionally frozen in those models, now the entire “backbone" is learnable. Also, where previously, the goal was to “map" vision to language, now the two are learned jointly. We therefore take this opportunity to investigate the models to better understand how their behaviors differ due to the two (pre-)training objectives. For a fair comparison with ViLT, we use **VLC**-Base which is trained with the same model architecture and image-text pairs.



Original Image

ViLT clusters

VLC clusters

Figure 4.5: Visualization of patch clusters for an example image as produced from ViLT (many densely clustered patches) versus **VLC**'s more fine-grained and diffuse representations. We believe this representational difference makes for easier and faster learning and scaling – akin to "fast mapping" in language acquisition.

Model	Image Size	Top-1 (Base)	Top-1 (Large)
Supervised			
ViT-B/16 [66]	384^2	77.9	76.5
DeiT-B [229]	384^2	83.1	-
Swin-B [157]	384^2	**84.5**	-
Self-supervised			
DINO [31]	224^2	82.8	-
MoCo v3 [47]	224^2	83.2	84.1
MaskFeat [246]	224^2	83.6	85.7
SimMIM [258]	224^2	83.8	85.4
BEiT* [19]	384^2	**84.6**	85.2
MAE [91]	224^2	83.6	85.9
VLC (ours)	384^2	**84.5**	**86.3**

Table 4.3: Models are pretrained on ImageNet 1K and self-supervised models are evaluated by end-to-end fine-tuning. *BEiT uses a DALLE [192] pre-trained tokenizer.

Understanding Patches. We begin with a simple patch clustering visualization (Figure 4.5). Without the inclusion of any language, we can simply cluster (and color) the visual patch embeddings of ViLT and **VLC**. ViLT relies on on larger patches (32×32) for higher resolution (384×640). We instead use smaller patches and lower resolution (16×16 for 384×384). It is easy to see how both models are identifying key semantic regions of the image (e.g. the rug, painting and plant). Also note, both models incorrectly place the painting and plant in the same cluster.

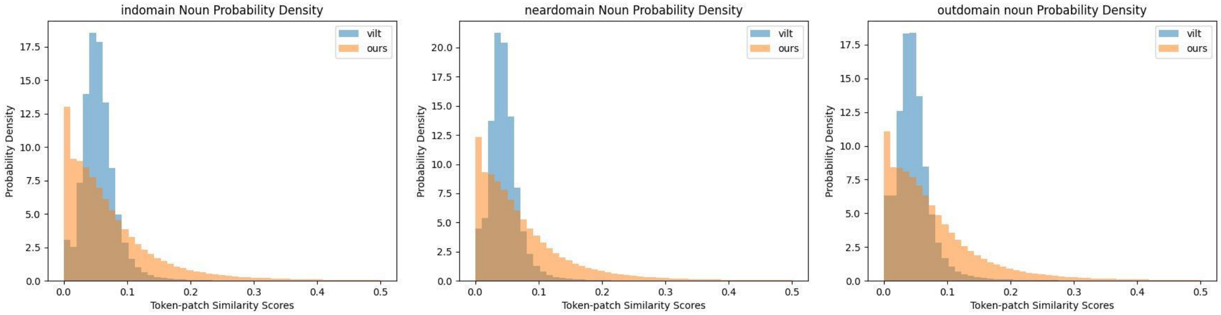


Figure 4.6: Plots of the top noun-patch similarity per image for ViLT and **VLC**-Base. ViLT rarely produces a high similarity lexical score, likely due to its discriminative pretraining objective and its score distribution shifts down as we move further away from its supervised pretraining data. In contrast, **VLC**-Base has a smoother distribution and high lexical alignment across all settings.

To investigate this representation collapse at scale, we leverage the nocaps dataset [2]. Nocaps provides captions for images based on object classes in COCO, similar to COCO, and out of domain. By visualizing the embedding similarities of nouns from these three classes with patches in the images, we can determine: 1. Are ViLT patches more tightly clustered – perhaps due to the discriminative training objective and 2. How do both models' behaviors change for classes more (or less) like the ImageNet pretraining. In Figure 4.6, we see several trends. First, ViLT's "most similar" patch to the noun rarely has a passes 0.1, perhaps indicating that they are not shifting from their pretrained representations. Second, we see the mass shift slightly lower as we move from left to right (in-domain to out-of-domain), indicating the model has a harder time finding alignments to novel words. **VLC** has a markedly different behavior, with a smoother overall set of similarities – often able to to find a visual patch with high similarity to the query across all conditions. **VLC** also exhibits an opposite trend where the model's scores climb as we shift out of domain. These plots do *not* show if the alignment is semantically meaningful, but they do show starkly different behaviors. This concentration of embeddings by ViLT can also be seen visually in examples in the Appendix A.4.

Image Classification. Given that the underlying visual representations are shifting through the cross-modal training, we run a simple image classification experiment to see the effects language training has on the underlying visual "backbone". We compare **VLC** with state-of-the-art models on ImageNet-1K classification and report top-1 validation accuracy of a single 384×384 crop.

As shown in Table 4.3, **VLC** learns generic representations which are transferable to vision tasks. With only fine-tuning on ImageNet-1K, our model matches the performance of Swin-B [157] that is trained with supervised labels. Note that BEiT [19] is a two-stage pre-training

	Model	Model Size	MC-test	DA-test
	Random	0	25.36	0.06
Large-scale pretrained model	BERT [62]	110M	33.54	8.41
	GPT-3 [27]	175B	35.21	11.49
	ResNet [89]	23M	28.81	2.30
	CLIP [189]	150M	**51.01**	7.10
Specialized model	ViLBERT [159]	274M	41.5	25.9
	LXMERT [223]	240M	41.6	25.9
	KRISP [167]	300M	42.2	27.1
	VLC-Base (ours)	195M	44.82	**27.49**
	ClipCap[†] [174]	930M	51.43	25.90
	GPV-2[‡] [109]	380M	53.7	40.7

Table 4.4: We compare **VLC** with state-of-the-art methods on A-OKVQA dataset in both Multiple-Choice (MC) and Direct Answer (DA) evaluation settings. [†]ClipCap uses pretrained CLIP and GPT-2 large as the encoder and decoder. [‡]GPV-2 uses pretrained VinVL and T5-base and learns a large number of concepts with Bing data. We report accuracy (%) on the test split returned from the evaluation server.

model of which the tokenizer is trained on 250M examples of DALLE [192] data. Compared with MAE [91], our model learns competitive multi-modal representations from vision-language pre-training while retains high-quality image representations.

Evaluation on Open-domain VQA. To investigate if the alignments between image patches and text tokens are semantically meaningful, we evaluate our **VLC** on A-OKVQA dataset [206]. Different from VQA [80], A-OKVQA requires some form of commonsense reasoning about the scene depicted in the image. In *multiple choice* (MC) setting, a model chooses its answer from one of four options. In the *direct answer* setting, a model can generate any text as its answer that is more applicable in real-world scenarios. We use **VLC** as the mulitmodal encoder and a pre-trained BERT to generate answers. In Table 4.4, we compare **VLC** with large-scale pretrained discriminative models (BERT [62], ResNet [89]), contrastive model (CLIP [189]), generative model (GPT-3 [27]) and models specifically designed for open-domain VQA tasks. While CLIP trained with $400M$ image-text pairs is very strong for multiple choice matching, it performs

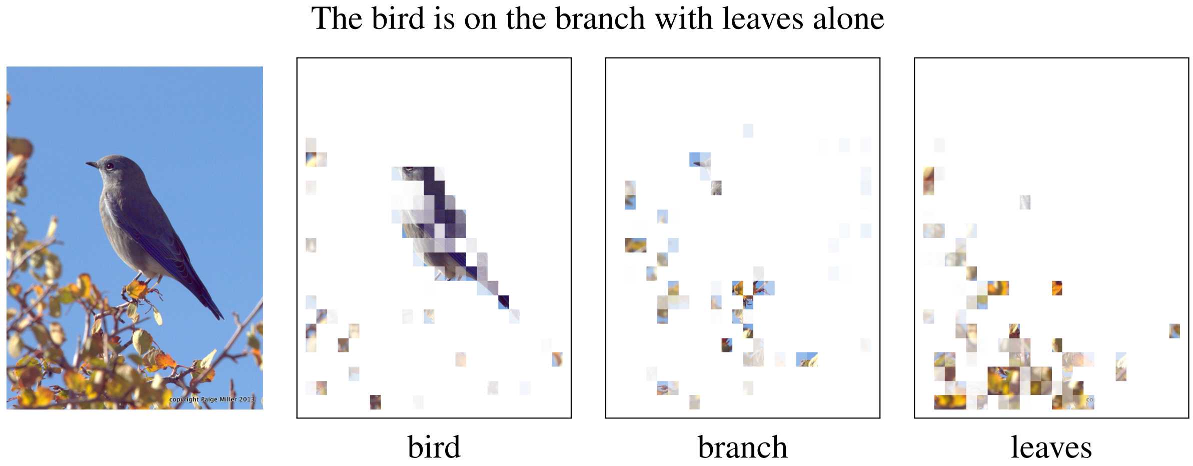


Figure 4.7: Lexical-Patch alignment for an COCO image. We visualize three different words from the same caption to see how the model uniquely represents them. This is a particularly challenging case as the model attempts to differentiate branches from leaves.

worse than other baselines in the DA setting. KRISP [167] ensembles different pretrained image classification and object detection models to exact image features. As a comparison, our **VLC** outperforms KRISP in both settings. It implies that our model provides more powerful image features by aligning image patches and text tokens. Note that ClipCap [174] and GPV-2 [109] use either much more data for pretraining or finetuning on the open-domain VQA task.

Visualizations. These patch-language transformer architectures allow for intuitive visualizations of the lexical alignment. Doing so provides a simple way to explore what the model is learning to represent about an image. In Figure 4.7, we show results from visualizing three different words in the same caption for an image from COCO. Note that for the word branch, the model is actively attempting to avoid the abundant leaves. Second, since there is nothing about our model besides the MAE initialization that should be biased (as shown previously) towards ImageNet classes, we present three images in Figure 5.4 that highlight words not present in the standard ImageNet1K training split used by other models. Specifically, a noun (*string*), adjective (*yellow*), and verb (*swinging*). These demonstrate the general trend of ViLT often focusing on surprising locations. We show additional examples for nouns in Figure 4.9, adjectives in Figure 4.10, and verbs in Figure 4.11.

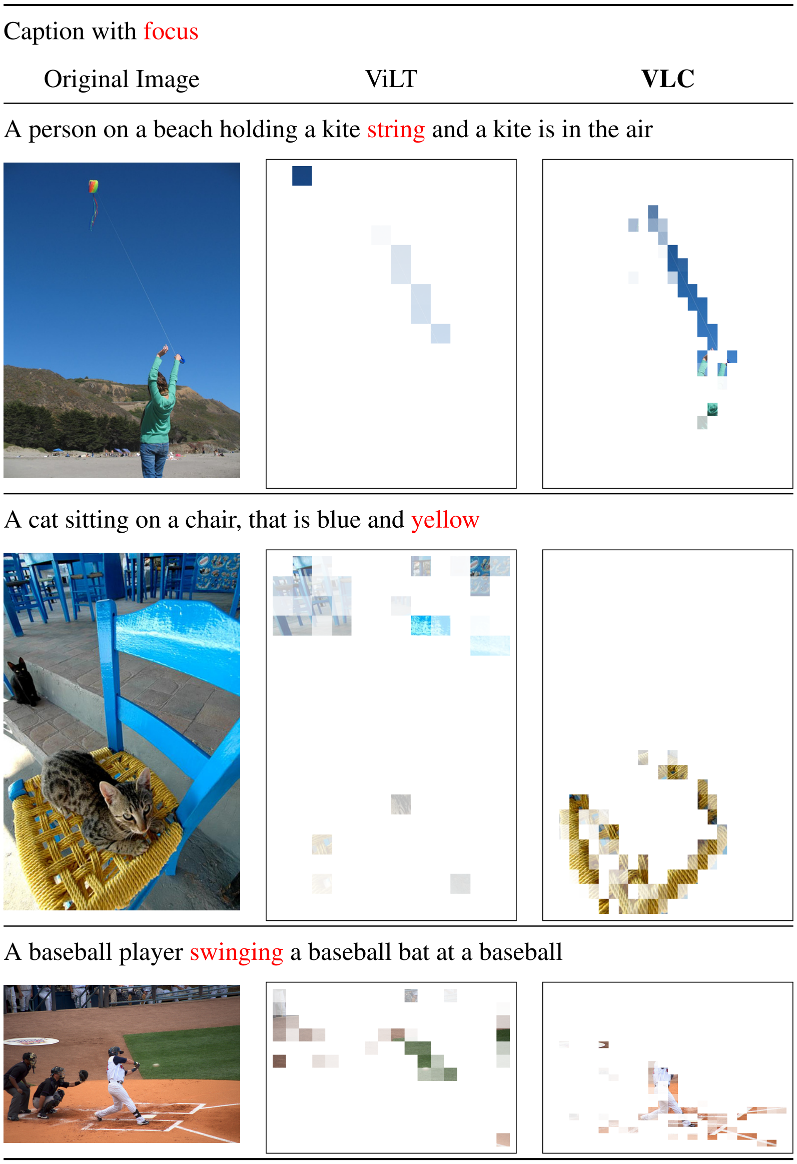

Caption with focus
Original Image
ViLT
VLC
A person on a beach holding a kite string and a kite is in the air
A cat sitting on a chair, that is blue and yellow
A baseball player swinging a baseball bat at a baseball

Figure 4.8: To investigate concepts not present in COCO or ImageNet, we present three images and highlighted words which are out of domain (i.e. not in ImageNet-1K). Specifically, we are visualizing a noun (top), adjective (middle) and verb (bottom). The model again delicately avoids nearby but distinct concepts (e.g. the cat on the chair or irrelevant parts of the baseball field).

4.7 Summary

We present **V**ision-**L**anguage from **C**aptions (**VLC**), a transformer pretrained with *only* image-caption pairs. While **VLC** uses only a linear projection layer for image embedding, it achieves competitive performance on a diverse set of vision-language tasks as compared to existing approaches that rely on object detectors or supervised CNN/ViT networks. We perform a number

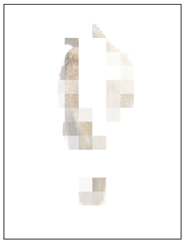
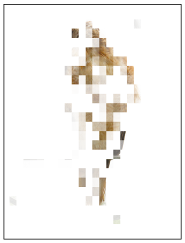
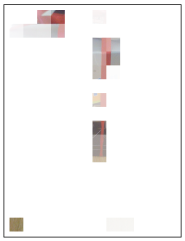
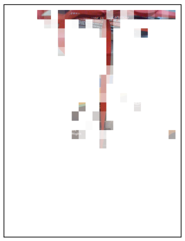
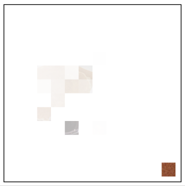
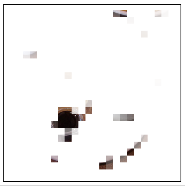
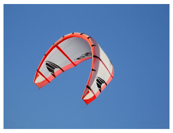
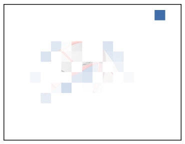
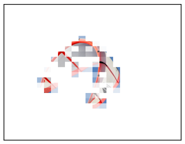
Caption with focus	Original Image	ViLT	**VLC**
A hawk is perched on a metal bar			
A gift wrapped with a ribbon sits on a table with a knife			
A plate with pancakes, syrup, grits, and butter			
There is a colorful parachute in the sky			

Figure 4.9: Visualized are OOD noun examples. Note that ViLT is often picking up on relevant features but has a single strongest correlation with a single, presumably predictive, patch.

of analysis and investigations of the representations. For example, we demonstrate **VLC** visual representations are effective for ImageNet-1K classification and our visualization demonstrates that **VLC** can accurately align image patches with text tokens. As performance scales with increased training data this opens an exciting avenue for large-scale weakly-supervised open-domain vision-language models.

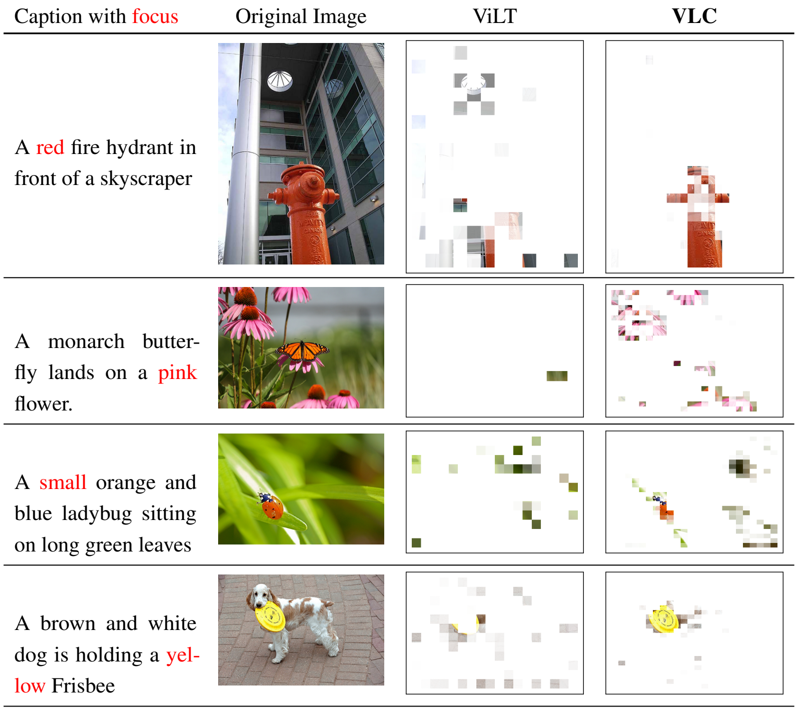

Caption with focus
Original Image
ViLT
VLC
A red fire hydrant in front of a skyscraper
A monarch butterfly lands on a pink flower.
A small orange and blue ladybug sitting on long green leaves
A brown and white dog is holding a yellow Frisbee

Figure 4.10: Visualized are OOD adjective examples. **VLC** produces more accurate and comprehensive masks. Note that the lady bug is correctly identified but not exclusively and likely not based on an understanding of the relative size *small*. Future work would ideally show results that indicate models understanding more abstract and comparative concepts.

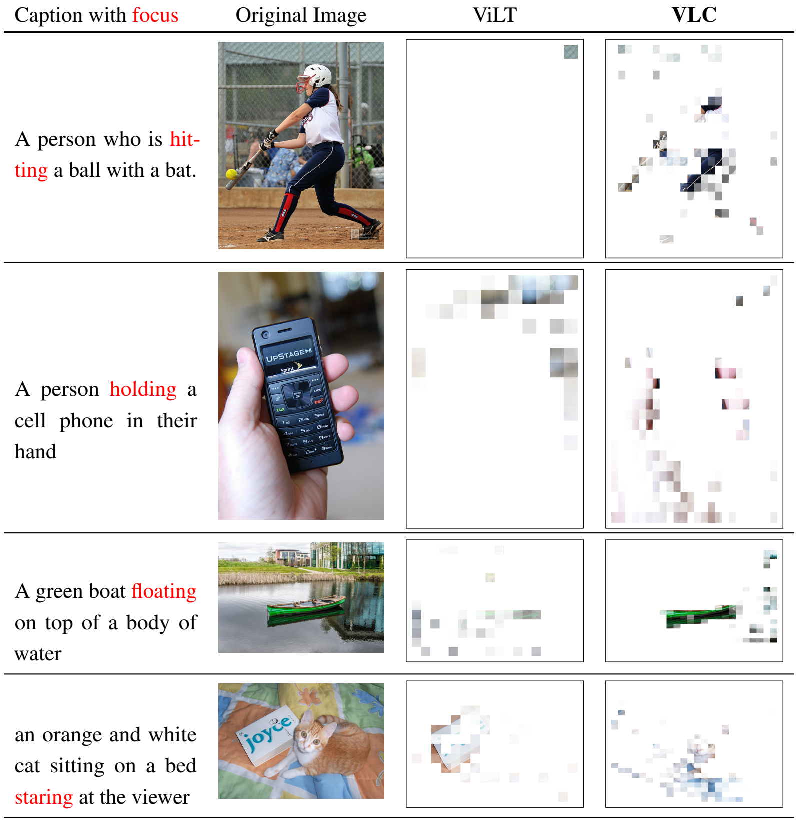

Caption with focus
Original Image
ViLT
VLC
A person who is hitting a ball with a bat.
A person holding a cell phone in their hand
A green boat floating on top of a body of water
an orange and white cat sitting on a bed staring at the viewer

Figure 4.11: Visualized are OOD verb examples. Note that verbs from still images is a slightly strange concept, but there are key perceptual indicators that align to the verb's semantics. For example, *holding* is aligned to the person's hands and *staring* picks up on the cat's eyes.

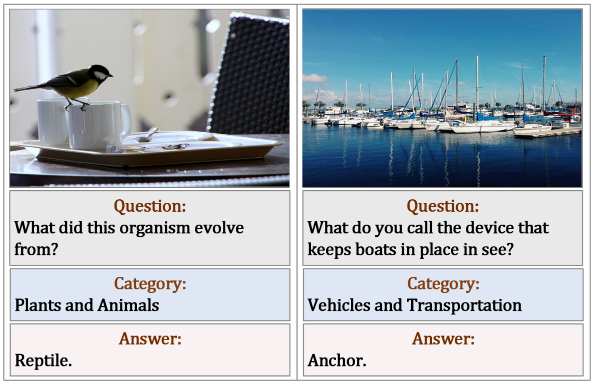
Chapter 5

Open-Ended Visual Question Answering with External and Implicit Knowledge

In this chapter, we explore the integration of external knowledge bases and the implicit knowledge embedded in large language models to address the limitations of human-annotated datasets in open-ended visual question answering. By leveraging these auxiliary sources of knowledge, the proposed method improves answering capabilities while significantly reducing the reliance on human-generated annotations.

5.1 Overview

The primary focus of recent work with large-scale transformers has been on optimizing the amount of information packed into the model's parameters. In this work, we ask a complementary question: Can multimodal transformers leverage explicit knowledge in their reasoning? Existing, primarily unimodal, methods have explored approaches under the paradigm of knowledge retrieval followed by answer prediction, but leave open questions about the quality and relevance of the retrieved knowledge used, and how the reasoning processes over implicit and explicit knowledge should be integrated. To address these challenges, we propose a - **K**nowledge **A**ugmented **T**ransformer (KAT) - which achieves a strong state-of-the-art result (+6% absolute) on the open-domain multimodal task of OK-VQA. Our approach integrates implicit and explicit knowledge in an encoder-decoder architecture, while still jointly reasoning over both knowledge sources during answer generation. Additionally, explicit knowledge integration improves interpretability of model predictions in our analysis.



Question: What did this organism evolve from?	Question: What do you call the device that keeps boats in place in see?
Category: Plants and Animals	Category: Vehicles and Transportation
Answer: Reptile.	Answer: Anchor.

Figure 5.1: Examples of knowledge-based VQA that requires external knowledge. Success on this task requires not only visual recognition, but also logical reasoning to incorporate external knowledge about the world.

5.2 Motivation

There has been a revival of interest in knowledge-intensive tasks which require an external knowledge source for humans to perform. Many applications in real-world scenarios, such as autonomous AI agents, need to seamlessly integrate implicit (*i.e.*, commonsense) and explicit knowledge (*e.g.*, Wikidata) to answer questions. In this work, we investigate how to effectively integrate implicit and explicit knowledge for reasoning. Tasks like Outside Knowledge Visual Question Answering (OK-VQA) [166] require that models use knowledge not present in the input to answer questions, making it an ideal test bed for investigating this implicit-explicit knowledge trade-off.

Consider the examples from OK-VQA shown in Figure 5.1. To answer the question in the left example, the system needs to both ground *organism* to bird through explicit knowledge and then apply the implicit knowledge *birds evolved from reptiles* to answer the question. Similarly for the question in the right example, the system needs to recognize boats and harbor and requires the implicit knowledge *anchors are used to stop boats from moving*. A key challenge here is to accurately link image content to abstract external knowledge. There have been a number of recent developments demonstrating the feasibility of incorporating external knowledge into Question Answering models [75, 130, 167, 238, 250]. Existing methods first retrieve external knowledge from external knowledge resources, such as DBPedia [13] and ConceptNet [154] before jointly reasoning over the retrieved knowledge and image content to predict an answer.

However, most existing approaches have several drawbacks. First, explicit knowledge retrieved using keywords from questions or image tags may be too generic, which leads noise or irrelevant knowledge during knowledge reasoning. Second, existing work mainly focuses on explicit knowledge which is often in the form of encyclopedia articles or knowledge graphs. While this type of knowledge can be useful, it is insufficient to answer many knowledge-based questions. As shown in Figure 5.1, questions require the system to jointly reason over explicit and implicit knowledge, which is analogous to the way humans do.

To address these challenges, we propose an approach, **KAT**, to effectively integrate implicit and explicit knowledge during reasoning. The main contributions of our work are as follows:
i) Knowledge extraction. We adopt two novel methods for knowledge extraction that significantly improve the quality and relevance of extracted knowledge: for implicit knowledge, we design new prompts to extract both tentative answers and supporting evidence from a frozen GPT-3 model; for explicit knowledge, we design a contrastive-learning-based explicit knowledge retriever using the CLIP model, where all the retrieved knowledge are centered around visually-aligned entities.
ii) Reasoning in an encoder-decoder transformer. We design a novel reasoning module in *KAT* to perform jointly reasoning over explicit and implicit knowledge during answer generation, which is trained by using an end-to-end encoder-decoder transformer architecture.
iii) OK-VQA performance. *KAT* sets a new state of the art on the challenging OK-VQA [166] benchmark, and significantly outperforms existing approaches.

5.3 Prior Work

Vision-Language Transformer. Multimodal transformers have made significant progress over the past few years, by pre-trained on large-scale image and text pairs, then finetuned on downstream tasks. VisualBERT [139], Unicoder-VL [129], NICE [40], and VL-BERT [214] propose the single-stream architecture to work on both images and text. ViLBERT [159] and LXMERT [223] propose a two-stream architecture to process images and text independently and fused by a third transformer in ta later stage. While these models have shown to store in-depth cross-modal knowledge and achieved impressive performance on knowledge-based VQA [160, 167, 250], this type of implicitly learned knowledge is not sufficient to answer many knowledge-based questions [167]. Another line of work for multimodal transformers, such as CLIP [189] or ALIGN [103], aligns visual and language representations by contrastive learning. These models achieve state-of-the-art performance on image-text retrieval tasks. Different from existing work that uses multimodal transformers as implicit knowledge bases, we focus primarily on how to associate images with external knowledge. Importantly, our model only relies on multimodal transformers learned by contrastive learning which do not require any labeled images. This makes our model more flexible in real-world scenarios.

Knowledge-based VQA. Some Knowledge-based visual language tasks requires external knowledge beyond the image to answer a question. Early exploration, such as FVQA [237], creates a fact-based VQA dataset by selecting a fact (*e.g., <Cat, CapableOf, ClimbingTrees>*) from a fixed knowledge base. A recent Outside Knowledge VQA (OK-VQA) dataset is a more challenging dataset, covering a wide range of knowledge categories. In our work, we focus on OK-VQA due to its large-scale knowledge-based questions as well as its open-ended nature.

Recent approaches have shown a great potential to incorporate external knowledge for knowledge-based VQA. Several methods explore aggregating the external knowledge either in the form of structured knowledge graphs [75, 130, 175, 237, 238], unstructured knowledge bases [160, 167, 250], and neural-symbolic inference based knowledge [38, 248]. In these methods, object detectors [196] and scene classifiers [89] are used to associate images with external knowledge. Further, external APIs, such as Google [160, 250], Microsoft [39, 265] and OCR [160, 250] are used to enrich the associated knowledge. Finally, pre-trained transformer-based language models [39, 265], or multimodal models [75, 160, 167, 250, 250] are leveraged as implicit knowledge bases for answer predictions.

Different from previous approaches, Our work aims to develop a single, unified architecture, by jointly reasoning over explicit and implicit knowledge to augment generative language models. While part of our approach is similar to PICa [265] which considers GPT-3 as implicit knowledge base, our model takes one step further by showing that how explicit and implicit knowledge can be integrated during knowledge reasoning. Another similar work Vis-DPR [160] collects a knowledge corpus from training set by Google Search which is specific to a certain dataset. Our proposed model is more generic by collecting entities from Wikidata and not limited to the training set.

Open-Domain Question Answering (ODQA). ODQA is the NLP task of answering general domain questions, in which the evidence is not given as input to the system. Several approaches [34, 111] propose to predict the answers by first retrieving support document from Wikipedia, before extracting answers from the retrieved document. Recent works [101, 128] combine text retrieval models with language generative models which achieve state-of-the-art performance on knowledge-intensive natural language processing tasks. Similar to these works as part of our method, we extend this framework to VQA domain and show the effectiveness of aggregating explicit and implicit knowledge for knowledge-based VQA.

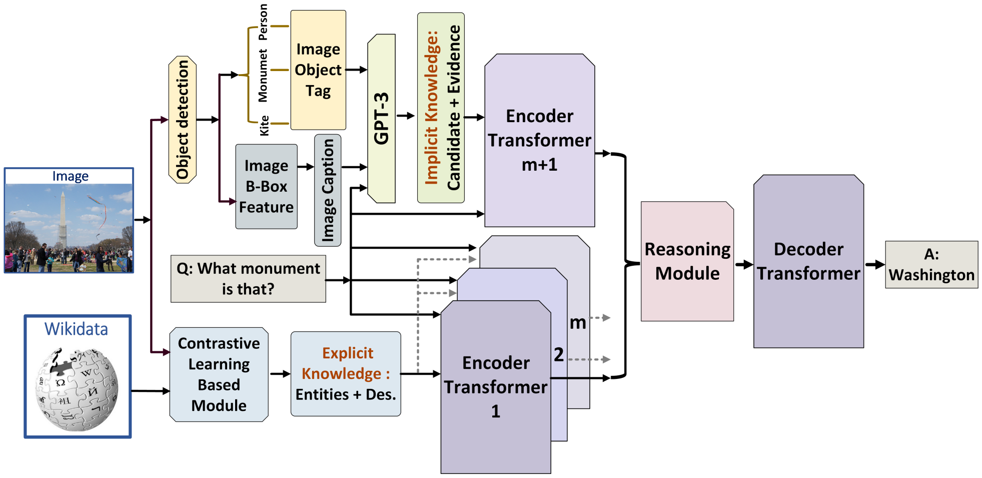


Figure 5.2: Our *KAT* model uses a contrastive-learning-based module to retrieve knowledge entries from an explicit knowledge base, and uses GPT-3 to retrieve implicit knowledge with supporting evidence. The integration of knowledge is processed by the respective encoder transformer, and jointly with reasoning module and the decoder transformer as an end-to-end training with the answer generation.

5.4 Knowledge Augmented Transformer

Overview

When humans reason about the world, they process multiple modalities and combine external and internal knowledge related to these inputs. Inspired by this idea, we introduce a new *KAT* approach. The overview of the proposed *KAT* model is shown in Figure 5.2. We define the knowledge from explicit knowledge bases as the explicit knowledge, and the knowledge stored in large-scale language models as the implicit knowledge (*i.e.*, implicit commonsense knowledge). We describe the retrieval method of our explicit knowledge (§5.4) and the retrieval method of our implicit knowledge (§5.4). Next, we introduce the details of our knowledge reasoning module which jointly reasons over both explicit and implicit knowledge (§5.4).

Problem Formulation. We apply our *KAT* on OK-VQA task in this paper. Formally, given a training dataset $\mathbb{D} = \{(v_i, q_i, a_i)\}_{i=1}^{s}$, where v_i denotes the i^{th} training image; s is the total number of the training images; q_i and a_i represent the i^{th} question and its corresponding answer, respectively. We use a sequence-to-sequence model that is composed of an encoder and a decoder, which is a comparison method of T5 [191] or BART [127]. Let θ be the parameters of the model p that needs to be trained. Unlike previous approaches that treat this task as a classification problem [167, 250], our model is to take v_i and q_i as inputs and generate the answer a_i in an auto-regressive manner. It should be noted that our proposed model tackles a more challenging problem. As the generated answer may contain an arbitrary number of words from the entire vocabulary.

Explicit Knowledge Retrieval

Explicit Knowledge Extraction

Given an image v_i and corresponding question q_i, it is important to ground image regions with fine-grained descriptions, which is conducive to understanding both the image content and the question with the referred items. Existing approaches [103, 189] on OK-VQA apply object detectors to generate image tags which are used for explicit knowledge retrieval. Such image tags can be generic and have a limited vocabulary size, leading noise or irrelevant knowledge. Motivated by the recent progress of visual-semantic matching approaches [103, 189], we leverage a contrastive-learning-based model to associate image regions with external knowledge bases.

Similar to the previous work [160, 167] which uses a subset of external knowledge, we construct an explicit knowledge base that covers the 8 categories of animals, vehicles and other common objects from Wikidata [234]. The details can be found in Section 5.4. We denote the constructed knowledge base as $\mathcal{K}$. Each knowledge entry e from $\mathcal{K}$ is a concatenation of the entity and its corresponding description.

The goal of our explicit knowledge retriever is to index all knowledge entries in d_r-dimensional dense representations by a dense encoder $E_{ent}(\cdot)$, such that it can efficiently retrieve the top m knowledge entries relevant to each input image. Given an image v_i, we use a sliding window with a stride to generate N image regions $\{v_i^1, ..., v_i^N\}$. Then an image encoder $E_{img}(\cdot)$ is applied to map each patch to a d_r-dimensional dense representation, and retrieves k knowledge entries from $\mathcal{K}$ whose representations are closest to the patch-level representation. To define the similarity score between the image region v_i^j and the entity e, we use the inner product of their normalized representations:

$$sim(v_i^j, e) = E_{ent}(e)^T E_{img}(v_i^j). \tag{5.1}$$

In total, we retrieve the top $N \times k$ knowledge entries relevant to image v_i. We keep top-m knowledge entries ranked by similarity scores as explicit knowledge source x^{exp}.

In principle, the image and knowledge entry encoders can be implemented by any multimodal transformer. We use the CLIP model (ViT-B/16 variant) [189] in our work and take the `[CLS]` as representations. We pre-extract representations of the knowledge entries in the knowledge base $\mathcal{K}$ using the entity encoder E_{ent} and index them using FAISS [107].

Knowledge Base Construction

We use the English Wikidata [234] dump from Sep. 20, 2021 as the explicit knowledge source base which contains $95,870,584$ entities. Each data item is stored in a structured format constituted of property-value pairs. Properties are objects and have their own Wikidata pages with labels, aliases, and descriptions. We extract a subset that covers common objects in real-world scenarios. We remove all entities whose string labels or corresponding descriptions are empty or non-English. This results in a total of $423,520$ entity triplets in the end (*e.g.*, *<Q2813, Coca-Cola, carbonated brown colored soft drink>*) (See Table 5.1).

Subclass		Number
Role	(Q214339)	162,027
Point of interest	(Q960648)	85,900
Tool	(Q39546)	78,621
Vehicle	(Q42889)	44,274
Animal	(Q729)	18,581
Clothing	(Q11460)	17,711
Company	(Q891723)	12,173
Sport	(Q349)	4,233
Total		423,520

Table 5.1: We collect a subset of Wikidata that covers common objects in real-life scenarios as our explicit knowledge base. Above are statistics of these subclasses.

Implicit Knowledge Retrieval

While our explicit knowledge retriever focuses on semantic matching between image regions and knowledge entries, it lacks implicit commonsense knowledge (*e.g.*, *Lemons are sour*) which is usually stored in large-scale language models [27]. In this section, we retrieve implicit knowledge with supporting evidence by prompting from a large-scale pre-trained language model.

We design our implicit knowledge retriever with inspirations from the previous work [265]. We leverage GPT-3 as an implicit language knowledge base and treat VQA as an open-ended text generation task. For each image-question pair, we first convert the image v_i into a textual description C via a state-of-the-art image captioning model [143], and then construct a carefully designed text prompt consisting of a general instruction sentence, the textual description C, the question, and a set of context-question-answer triplets taken from the training dataset that are semantically most similar to the current image-question pair for a concrete example). We then input this text prompt to the GPT-3 model in its frozen version and obtain the output from GPT-3 as the tentative answer candidate to the current image-question pair.

To gain deeper insights from the implicit knowledge coming out of GPT-3 and its rationale, we design another prompt to query GPT-3 for supporting evidence behind the tentative answer candidate that it generates. More specifically, for each image-question pair (v_i, q_i), and for a tentative answer a generated by GPT-3, we construct the prompt in the form of: "(question q_i)? (answer a). This is because" to query GPT-3 for supporting evidence for a concrete example). We finally compile both the tentative answers and the corresponding supporting evidence from GPT-3 as implicit knowledge source x^{imp}.

KAT Model

As showed in the Figure 5.2, the explicit knowledge entries are from an image, which are concerned with semantic matching of the image regions. These knowledge entries could be noisy or irrelevant to its corresponding question. Moreover, some of the supporting evidence prompted from GPT-3 is generic or not related to image content. Simple concatenation of different knowledge may introduce noise during model training. We design a knowledge reasoning module with inspirations from the previous work [111]. Our knowledge reasoning module encodes each question and knowledge pair separately, and jointly reason over both explicit and implicit knowledge when generating an answer.

Encoder. We concatenate question q_i with each knowledge as a question-knowledge pair. Firstly, we add sentinel tokens `question:`, `entity:` and `description:` before the question, the retrieved entity, and its description separately. Similarly, we add sentinel tokens `question:`, `candidate:` and `evidence:` before the question, the tentative answer, and its evidence. Secondly, we use an embedding layer followed by a sequence of encoder layers to encode the question-knowledge pairs separately. We average the token embeddings of each question-knowledge pair from the last encoder layer, which results in an embedding matrix of explicit knowledge $X^{exp} \in \mathbb{R}^{m \times d}$ and implicit knowledge $X^{imp} \in \mathbb{R}^{p \times d}$, where d, m and p are the embedding dimension, the number of explicit knowledge x^{exp}, and the number of implicit knowledge x^{imp}, respectively.

Reasoning Module. To jointly reason over implicit and explicit knowledge, we concatenate the embeddings of explicit and implicit knowledge form a global representation $X \in \mathbb{R}^{(m+p) \times d}$. The cross-attention module takes the global representation X of the encoder as the input. Let $H \in \mathbb{R}^{d}$ be the output of the previous self-attention layer of the decoder. By definition [232], the scaled dot-product attention can be expressed as:

$$Q_v = softmax(\frac{QK^T}{\sqrt{d}})V, \tag{5.2}$$

where queries Q, keys K, and values V are computed by applying linear transformations: $Q = W_Q H, K = W_K X, V = W_V X$. The attended representation Q_v is a weighted sum of the values, and implies that our model performs a joint reasoning over explicit and implicit knowledge when generating answers.

Decoder. We feed the embeddings of explicit and implicit knowledge to a sequence of decoder layers for answer generation. We train our model with a cross-entropy loss:

$$\mathcal{L}_{CE} = -\sum_{t=1}^{n} \log p_\theta(y_t | y_{<t}, x^{exp}; x^{imp}), \tag{5.3}$$

where y_t is predicted autoregressively.

5.5 Empirical Evaluation

Dataset

OK-VQA [166] is currently the largest knowledge-based VQA dataset, The questions are crowd-sourced from Amazon Mechanical Turkers and require outside knowledge beyond the images in order to be answered correctly. The dataset contains $14,031$ images and $14,055$ questions covering a variety of knowledge categories. We follow the standard evaluation metric recommended by the VQA challenge [9].

Implementation Details

For the knowledge reasoning module, we initialize our model with the pre-trained T5 model [191]. We compare two model sizes, base and large, each containing $220M$ and $770M$ parameters respectively. We fine-tune the models on OK-VQA dataset, using AdamW [158]. We use a learning rate of $3e-5$ to warm up for $2K$ iterations and train for $10K$ iterations. Limited by the computational resources, we set the number of retrieved entities to 40. The model is trained with a batch size of 32, using 16 V100 GPUs with 32Gb of memory each. Unless otherwise specified, all results reported in this paper as KAT use this model which we found to perform best. We evaluate our predictions with ground-truth after normalization. The normalization step consists of lowercasing, and removing articles, punctuation and duplicated whitespace [34, 125]. To be consistent with previous work [167], we train our model with 3 different random seeds and use the average results for the leaderboard submission.

	Method	Knowledge Resources	Acc (%)
No knowledge	Q only [166]	-	14.93
	Vanilla T5	-	18.56
	MLP [166]	-	20.67
	BAN [166]	-	25.1
	MUTAN [166]	-	26.41
With knowledge	BAN+AN [166]	Wikipedia	25.61
	BAN+KG-AUG [130]	Wikipedia+ConceptNet	26.71
	MUTAN+AN [166]	Wikipedia	27.84
	ConceptBERT [75]	ConceptNet	33.66
	KRISP [167]	Wikipedia+ConceptNet	38.35
	Vis-DPR [160]	Google Search	39.2
	MAVEx [250]	Wikipedia+ConceptNet+Google Images	39.4
GPT-3	PICa-Base [265]	Frozen GPT-3 (175B)	43.3
	PICa-Full [265]	Frozen GPT-3 (175B)	48.0
	KAT-explicit (w/ reasoning)	Wikidata	44.25
	KAT-implicit (w/ reasoning)	Frozen GPT-3 (175B)	49.72
	KAT (w/o reasoning)	Wikidata+Frozen GPT-3 (175B)	51.97
	KAT (single)	Wikidata+Frozen GPT-3 (175B)	53.09
	KAT (ensemble)	Wikidata+Frozen GPT-3 (175B)	**54.41**

Table 5.2: Results of OK-VQA comparing to standard baselines show that our KAT (large size) model achieves state-of-the-art performance on OK-VQA full testing set. It is important (see table sections) to compare methods based on their access to increasingly large implicit sources of knowledge and utilization of explicit knowledge sources. Our five KAT models variants make the relative importance of these decisions explicit. We train our model with 3 random seeds and the result is denoted as *ensemble*.

Comparison with Existing Approaches

We compare our model against existing approaches on the OK-VQA dataset and the results are summarized in Table 5.2. Our model outperforms state-of-the-art methods by significant margins. We compare our model with existing approaches from two aspects. (1) If we only consider using explicit knowledge, our model achieves 44.25% which is 4.85% and 5.9% higher than MAVEx and KRISP, respectively. Our model uses contrastive-learning-based model to extract knowledge, leaving headroom by incorporating supervised pre-trained models, such as pre-trained object detectors. It should be noted that our proposed model is working on a more challenging problem. As the generated answer could contain an arbitrary number of words from the entire vocabulary. Our model is slightly better than PICa-Base which is a plain version of PICa-Full without example engineering. It implies that our single, unified architecture can effectively associate images with the explicit knowledge base. (2) If we take the implicit knowledge from GPT-3 as the additional input, our model outperforms PICa-Full by 6.41% which indicates it is important to integrate knowledge of different types when generating answers. The detailed comparison can be found in Table 5.3.

5.6 Understanding the Models

To unpack the performance gain and understand the impact of different components, we ablate and compare different model architectures, types of knowledge and the number of explicit knowledge.

Model architecture		**Knowledge**		**Accuracy (%)**
Base	Large	Explicit	Implicit	
√				18.56
√		√		40.93
	√	√		44.25
√			√	47.60
	√		√	49.72
√		√	√	50.58
	√	√	√	54.41

Table 5.3: Ablation study on model architectures and types of knowledge. Our experiments show that larger model has more capacity for implicit knowledge reasoning and jointly reasoning over both knowledge sources has a consistent improvement with baselines.

Specifically, as shown in Table 5.3, our KAT-large shows a consistent improvement over using KAT-base. This larger model has more capacity for implicit knowledge reasoning. The integration of explicit and implicit knowledge achieves a performance gain of $\sim 4\%$, supporting the intuition that these two types of knowledge provide complementary pieces of knowledge.

Effectiveness of Knowledge Reasoning

To verify the effectiveness of our knowledge reasoning module, we use a KAT without the knowledge reasoning module which is denoted as KAT (w/o reasoning). This model concatenates explicit and implicit knowledge as a sentence and adopts a maximum length of 256 tokens. We train this variant with the same parameter settings. As shown in Table 5.4, simply concatenating knowledge sources is 2.43% lower than our proposed model. It indicates that KAT (w/o reasoning)

Method	Accuracy (%)
KAT (w/o reasoning)	51.97
KAT	**54.41**

Table 5.4: Comparison with KAT (w/o reasoning) which uses the concatenated knowledge as inputs without the knowledge reasoning module.

may introduce noise to relevant knowledge during encoding. Our model adaptively attend different knowledge sources for answer generation that can reduce the influence of irrelevant knowledge.

Extracting Explicit Knowledge

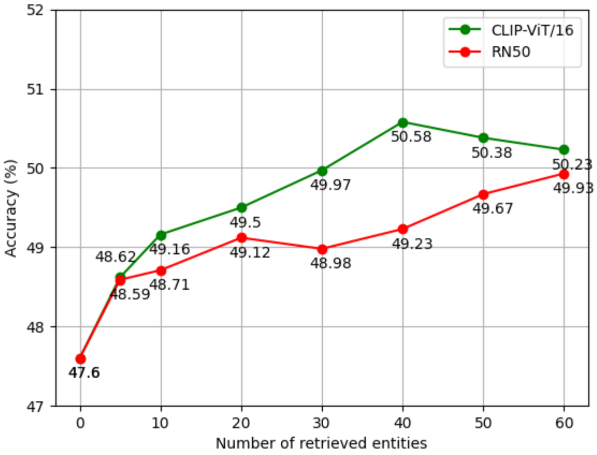


Figure 5.3: Our model achieves consistent improvement when aggregating more knowledge entries from an explicit knowledge base. However, as CLIP-ViT/16 and RN50 are very different explicit knowledge retrieval backbones we see the choice of backbone and number of sources to include are intimately related. Here we use KAT-base for demonstration.

From Figure 5.3 we can see, the performance of our model is directly affected by the size of retrieved explicit knowledge. When only considering the implicit knowledge (*i.e.*, the number of retrieved entities is 0), our model achieves 47.6% which is slightly worse than PICa-Full baseline. It indicates that solely increasing model complexity cannot improve the performance. This also demonstrates the importance of explicit knowledge. Our model shows a consistent improvement by incorporating more explicit knowledge. While a more extensive knowledge set may include more distracting knowledge, retrieved knowledge entries can share either visually or semantically similar knowledge as the relevant ones. Thus this can massively reduce the search space and/or reduce spurious ambiguity.

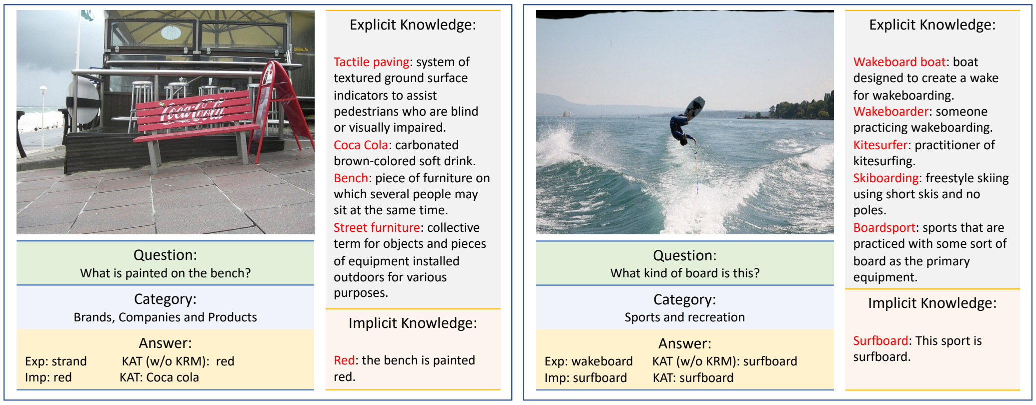

Explicit Knowledge:
Tactile paving: system of textured ground surface indicators to assist pedestrians who are blind or visually impaired.
Coca Cola: carbonated brown-colored soft drink.
Bench: piece of furniture on which several people may sit at the same time.
Street furniture: collective term for objects and pieces of equipment installed outdoors for various purposes.
Implicit Knowledge:
Red: the bench is painted red.
Question:
What is painted on the bench?
Category:
Brands, Companies and Products
Answer:
Exp: strand
Imp: red
KAT (w/o KRM): red
KAT: Coca cola
Explicit Knowledge:
Wakeboard boat: boat designed to create a wake for wakeboarding.
Wakeboarder: someone practicing wakeboarding.
Kitesurfer: practitioner of kitesurfing.
Skiboarding: freestyle skiing using short skis and no poles.
Boardsport: sports that are practiced with some sort of board as the primary equipment.
Implicit Knowledge:
Surfboard: This sport is surfboard.
Question:
What kind of board is this?
Category:
Sports and recreation
Answer:
Exp: wakeboard
Imp: surfboard
KAT (w/o KRM): surfboard
KAT: surfboard

Figure 5.4: Two examples from OK-VQA dataset that our model generates correct answers by jointly reasoning over both implicit and explicit knowledge. (exp: predictions by using explicit knowledge only and imp: predictions by using implicit knowledge only).

We compare different explicit knowledge retrieval module. Though ViT/16 has a large classification improvement over ResNet-50 (*e.g.*, 6.9% on ImageNet) [189], there is a less gap between these two backbones. As the number of retrieved entities increases, our knowledge reasoning module can further migrate this gap by adaptively attending to different explicit knowledge.

Category Results on OK-VQA

Here we present quantitative analyses to illustrate how explicit and implicit knowledge influence the final predictions. Based on the types of knowledge required, questions in OK-VQA are categorized into 11 categories and the accuracy results of each category are reported in Table 5.5. We re-train our model under the same settings with only either explicit or implicit knowledge, denoted as “exp” and “imp” respectively.

For most categories, the model using only explicit knowledge performs worse than that using only implicit knowledge. As implicit knowledge comes from the results of state-of-the-art object detection, image captioning models and supporting evidence by prompting GPT-3. While explicit knowledge is retrieved based on semantic matching between images and entities from knowledge bases, it contains richer but more distracting knowledge. Note that using explicit knowledge performs better for category “Brands, Companies, and Products" and “Weather and Climate". It indicates that accurately recognizing objects with fine-grained descriptions in the images is important for these categories to answer corresponding questions.

Question Type	Exp	Imp	**Ours**	Δ
Plants and Animals	42.2	51.5	54.7	+3.2
Science and Technology	44.4	43.3	52.8	+8.3
Sports and Recreation	49.7	53.8	60.4	+6.7
Geo, History, Lang, and Culture	45.6	45.4	55.8	+10.2
Brands, Companies, and Products	41.7	38.2	48.5	+6.8
Vehicles and Transportation	41.5	42.9	51.3	+8.4
Cooking and Food	47.9	47.7	52.7	+4.8
Weather and Climate	51.7	46.3	54.8	+3.1
People and Everyday	43.1	44.4	51.5	+7.1
Objects, Material and Clothing	42.9	45.4	49.3	+3.9
Other	41.5	50.2	51.2	+1.0

Table 5.5: Accuracy (%) of question types in OK-VQA full testing set. Our models outperforms exp and imp models by a large margin on all categories. (exp: explicit-only model and imp: implicit-only model)

Qualitative Analysis

Analyzed in previous sections, jointly reasoning over both knowledge sources during answer generation improves the explicit-only and implicit-only models by large margins. Figure 5.4 shows two examples comparing answers generated by different models along with retrieved knowledge. The left example shows that while explicit knowledge retrieved from the knowledge base contains the necessary knowledge entries for reasoning, it fails to generate the answer which requires the relation between bench and Coca Cola logos. On the other side, implicit knowledge retrieved from GPT-3 can only infer the bench is painted red, failing to recognize its logo. By jointly considering both knowledge sources, our model can associate the color of Coca Cola logo with the painted color of the bench which derives the correct answer. The right example shows that though explicit knowledge does not contain the right knowledge entries, it provides visually similar descriptions of this sport which further constrains the search space of our model and verifies the correctness of the implicit knowledge.

Analysis on More Examples

In this section, we showcase more predictions from variants of our model. As shown in Figure 5.5, we analyze the predictions based on different type of knowledge from several aspects:

Effectiveness of explicit knowledge retriever. Our explicit knowledge retriever can retrieve fine-grained knowledge entries from the explicit knowledge base, such as *golden retriever* (a fine-grained breed of dogs), *cucumber sandwich* (a specific type of sandwich) and *Macbook Pro* (a specific model of Apple products). These fine-grained entities are hardly obtained from existing object detection models, which can constraint the search space of our model and are beneficial to our answer generation process.

Effectiveness of implicit knowledge retriever. Our implicit knowledge retriever can retrieve supporting evidence from GPT-3, such as *Thomas: the train is named after the man who designed it.* and *Refrigerator: the refrigerator is used to keep food cold.* These kinds of knowledge are highly related to commonsense knowledge which needs further inference based on entities and provide complementary explanation to explicit knowledge.

Answer generation & classification. As most previous work on OK-VQA task, such as KRISP or MAVEx method, implement OK-VQA as a classification task. The prediction vocabulary is dataset-specific and assumes the training and test set are sharing a similar vocabulary. The limitation of these methods is the generalization ability. Our proposed KAT model treats OK-VQA as an open-end generation task. From these examples we found, our model can generate answers like *Iphone* or *Hercules* that are visually and semantically reasonable. Our proposed novel KAT model using the explicit and implicit knowledge is designed to enhance semantic alignment and generate representations with stronger knowledge-awareness.

5.7 Summary

This paper takes a step towards understanding the complementary role of implicit knowledge gained from continuing to scale models and explicit knowledge from structured knowledge bases. Importantly, it appears that there is headroom in both directions (i.g. improving retrieval and reasoning). Our conceptually simple yet effective approach for knowledge-based VQA makes these relationships explicit while still achieving a significant improvement against state-of-the-art results. Additional challenges remain, for example how best to align image regions with meaningful external semantics deserves and how to efficiently and accurately integrate multiple knowledge bases.

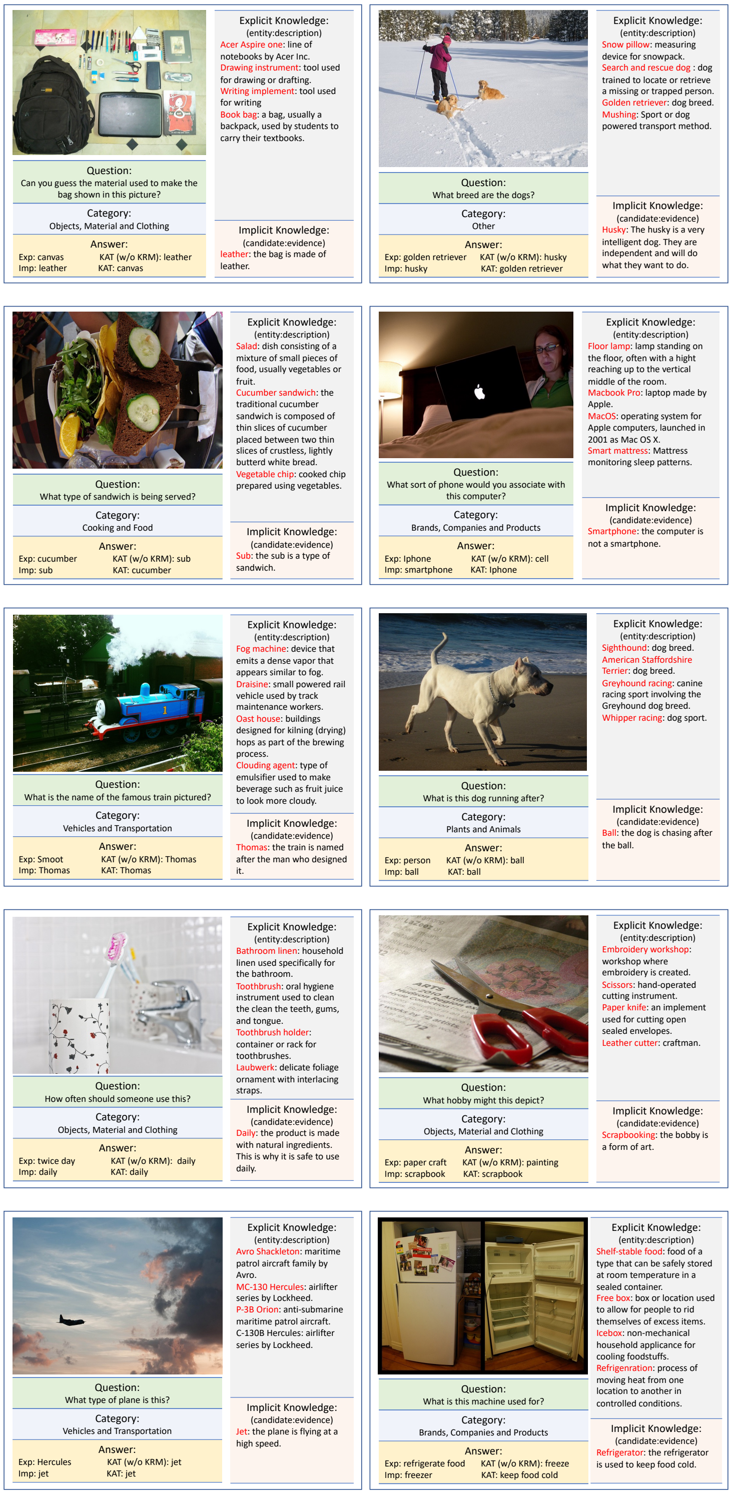

Explicit Knowledge:
(entity:description)
Acer Aspire one: line of notebooks by Acer Inc.
Drawing instrument: tool used for drawing or drafting.
Writing implement: tool used for writing
Book bag: a bag, usually a backpack, used by students to carry their textbooks.
Question:
Can you guess the material used to make the bag shown in this picture?
Category:
Objects, Material and Clothing
Answer:
Exp: canvas
KAT (w/o KRM): leather
Imp: leather
KAT: canvas
Implicit Knowledge:
(candidate:evidence)
leather: the bag is made of leather.
Explicit Knowledge:
(entity:description)
Snow pillow: measuring device for snowpack.
Search and rescue dog : dog trained to locate or retrieve a missing or trapped person.
Golden retriever: dog breed.
Mushing: Sport or dog powered transport method.
Question:
What breed are the dogs?
Category:
Other
Answer:
Exp: golden retriever
KAT (w/o KRM): husky
Imp: husky
KAT: golden retriever
Implicit Knowledge:
(candidate:evidence)
Husky: The husky is a very intelligent dog. They are independent and will do what they want to do.
Explicit Knowledge:
(entity:description)
Salad: dish consisting of a mixture of small pieces of food, usually vegetables or fruit.
Cucumber sandwich: the traditional cucumber sandwich is composed of thin slices of cucumber placed between two thin slices of crustless, lightly butterd white bread.
Vegetable chip: cooked chip prepared using vegetables.
Question:
What type of sandwich is being served?
Category:
Cooking and Food
Answer:
Exp: cucumber
KAT (w/o KRM): sub
Imp: sub
KAT: cucumber
Implicit Knowledge:
(candidate:evidence)
Sub: the sub is a type of sandwich.
Explicit Knowledge:
(entity:description)
Floor lamp: lamp standing on the floor, often with a hight reaching up to the vertical middle of the room.
Macbook Pro: laptop made by Apple.
MacOS: operating system for Apple computers, launched in 2001 as Mac OS X.
Smart mattress: Mattress monitoring sleep patterns.
Question:
What sort of phone would you associate with this computer?
Category:
Brands, Companies and Products
Answer:
Exp: Iphone
KAT (w/o KRM): cell
Imp: smartphone
KAT: Iphone
Implicit Knowledge:
(candidate:evidence)
Smartphone: the computer is not a smartphone.
Explicit Knowledge:
(entity:description)
Fog machine: device that emits a dense vapor that appears similar to fog.
Draisine: small powered rail vehicle used by track maintenance workers.
Oast house: buildings designed for kilning (drying) hops as part of the brewing process.
Clouding agent: type of emulsifier used to make beverage such as fruit juice to look more cloudy.
Question:
What is the name of the famous train pictured?
Category:
Vehicles and Transportation
Answer:
Exp: Smoot
KAT (w/o KRM): Thomas
Imp: Thomas
KAT: Thomas
Implicit Knowledge:
(candidate:evidence)
Thomas: the train is named after the man who designed it.
Explicit Knowledge:
(entity:description)
Sighthound: dog breed.
American Staffordshire Terrier: dog breed.
Greyhound racing: canine racing sport involving the Greyhound dog breed.
Whipper racing: dog sport.
Question:
What is this dog running after?
Category:
Plants and Animals
Answer:
Exp: person
KAT (w/o KRM): ball
Imp: ball
KAT: ball
Implicit Knowledge:
(candidate:evidence)
Ball: the dog is chasing after the ball.
Explicit Knowledge:
(entity:description)
Bathroom linen: household linen used specifically for the bathroom.
Toothbrush: oral hygiene instrument used to clean the clean the teeth, gums, and tongue.
Toothbrush holder: container or rack for toothbrushes.
Laubwerk: delicate foliage ornament with interlacing straps.
Question:
How often should someone use this?
Category:
Objects, Material and Clothing
Answer:
Exp: twice day
KAT (w/o KRM): daily
Imp: daily
KAT: daily
Implicit Knowledge:
(candidate:evidence)
Daily: the product is made with natural ingredients. This is why it is safe to use daily.
Explicit Knowledge:
(entity:description)
Embroidery workshop: workshop where embroidery is created.
Scissors: hand-operated cutting instrument.
Paper knife: an implement used for cutting open sealed envelopes.
Leather cutter: craftman.
Question:
What hobby might this depict?
Category:
Objects, Material and Clothing
Answer:
Exp: paper craft
KAT (w/o KRM): painting
Imp: scrapbook
KAT: scrapbook
Implicit Knowledge:
(candidate:evidence)
Scrapbooking: the bobby is a form of art.
Explicit Knowledge:
(entity:description)
Avro Shackleton: maritime patrol aircraft family by Avro.
MC-130 Hercules: airlifter series by Lockheed.
P-3B Orion: anti-submarine maritime patrol aircraft.
C-130B Hercules: airlifter series by Lockheed.
Question:
What type of plane is this?
Category:
Vehicles and Transportation
Answer:
Exp: Hercules
KAT (w/o KRM): jet
Imp: jet
KAT: jet
Implicit Knowledge:
(candidate:evidence)
Jet: the plane is flying at a high speed.
Explicit Knowledge:
(entity:description)
Shelf-stable food: food of a type that can be safely stored at room temperature in a sealed container.
Free box: box or location used to allow for people to rid themselves of excess items.
Icebox: non-mechanical household applicance for cooling foodstuffs.
Refrigenration: process of moving heat from one location to another in controlled conditions.
Question:
What is this machine used for?
Category:
Brands, Companies and Products
Answer:
Exp: refrigerate food
KAT (w/o KRM): freeze
Imp: freezer
KAT: keep food cold
Implicit Knowledge:
(candidate:evidence)
Refrigerator: the refrigerator is used to keep food cold.

Figure 5.5: More examples from OK-VQA dataset that our model generates answers by jointly reasoning over both implicit and explicit knowledge.

Part III

Preference Learning for Video Understanding

Chapter 6

Preference Learning with Large Language Models for Video Understanding

This chapter focuses on video-based tasks, including instruction following, captioning, and question answering, and introduces the method to utilize large language models for learning preferences. By aligning model behavior with implicit human preferences, the proposed approach reduces the reliance on annotated datasets while enhancing the effectiveness and generalization of video understanding models.

6.1 Overview

Preference modeling techniques, such as direct preference optimization (DPO), have shown to be effective in enhancing the generalization abilities of large language model (LLM). However, in tasks involving video instruction-following, providing informative feedback, especially for open-ended conversations, remains a significant challenge. Although previous studies have explored using large multimodal models (LMMs) as reward models to guide preference modeling, their ability to accurately assess the quality of generated responses and their alignment with video content has not been conclusively demonstrated. This paper introduces a novel framework that utilizes detailed video captions as a proxy of video content, enabling language models to incorporate this information as supporting evidence for scoring video question-answering (QA) predictions. Our approach demonstrates robust alignment with OpenAI GPT-4V model's reward mechanism, which directly takes video frames as input. Furthermore, we show that applying our reward mechanism to DPO algorithm significantly improves model performance on open-ended video QA tasks.

6.2 Motivation

This paper addresses the challenge of aligning LMMs, particularly in tasks that involve video instruction following. Despite recent advances in reinforcement learning (RL) [16, 124, 181, 220] and DPO [51, 94, 190], which have been effective in guiding LLMs towards generating more honest, helpful, and harmless content, their effectiveness in video domain remains limited. The critical obstacle lies in developing a robust reward system capable of distinguishing preferred responses from less preferred ones based on video inputs. The challenge is further complicated by the coverage and potential inaccuracies in generated content, stemming from the scarcity of alignment data across different modalities [153, 219].

Although human preference data are valuable, scaling is challenging due to its cost and labor intensive nature, as highlighted by the LLaVA-RLHF [219] paper, which collected 10k human-evaluated instances at a considerable cost of $3000. Existing approaches for distilling preferences, such as those for image data using GPT-4V [138], encounter scalability issues, especially for video inputs that require analyzing multiple frames. While [4] leverage a supervised finetuning (SFT) model for self-evaluation, the efficacy of the SFT model remains uncertain, particularly in accurately assessing the factuality of responses in relation to their corresponding videos.

To tackle the aforementioned challenges, we introduce a cost-effective reward mechanism that is both computationally and financially efficient for evaluating the quality of responses generated by video LLMs, serving as a basis for further on-policy preference optimization. We propose the use of detailed video captions as a proxy for video content, enabling a language model analyze the content and assess the quality of an LMM's response to related questions. The language model generates natural language feedback as a chain-of-thought step, and produces a numerical score as the reward, thereby creating an efficient feedback system.

However, high-quality video captions are essential for this process. To mitigate the shortage of high-quality video captions, we have developed a comprehensive video caption dataset, SHAREGPTVIDEO, using a simple prompting technique with the GPT-4V model, comprising 900k captions that encompass a wide range of video content, including temporal dynamics, world knowledge, object attributes, and spatial relationships. With this video caption dataset available, we verify that our reward mechanism, which utilizes video captions as a proxy, is well-aligned with evaluations derived from the more powerful, albeit costlier, GPT-4V model-generated rewards. Using this reward mechanism as the basis for the DPO algorithm, we train LLAVA-HOUND-DPO that achieves an improvement in accuracy of 8. 1% over the SFT counterpart. This marks a significant advancement in video LMM alignment and represents the first successful application of a DPO method in this domain.

Our contributions are outlined as follows:

1. We release a large-scale detailed video caption (900k) and instruction-following (900k)

dataset covering a wide range of video content, which facilitates video LMM model training and research.

2. We demonstrate the effective application of DPO to improve model performance by leveraging the language model feedback as reward, which substantially improves model performance on open-ended video QA tasks.
3. We propose an automated *development* benchmark for evaluating video instruction-following capability, serving as a cost-effective way to validate model performance.

6.3 Prior Work

Large Multi-Modal Models. LMMs [15, 37, 135, 152, 153] have enabled instruction following across modalities by utilizing LLM as backbones. In the context of video understanding, LLMs have been adapted to process video content [4, 106, 136, 155, 161, 163, 274]. Our work adopts Video-LLaVA [149] backbone, focusing on model enhancement through preference modeling with the DPO technique.

Video-text Datasets. Existing video-text datasets typically provide brief sentences or mere keywords as captions, as indicated by [18, 102, 241, 260, 270]. Video-ChatGPT [136] employs human effort to create high-quality video instructions, albeit limited to the ActivityNet domain with only 100k instruction pairs. Concurrent work [41] leverages GPT-4V to label video captions. Our work also leverages the GPT-4V model to produce detailed video captions, which we release as community resource for LMM training.

Preference Modeling for LMMs. Preference modeling techniques are DPO [87, 138, 219] or PPO [219] are applied to LMM alignment. More recently, [4] used RL on AI feedback to improve video LMM performance. Our contribution extends DPO to the video LMM alignment, with the use of detailed captions as factual evidence for reward modeling.

6.4 Method

As shown in fig. 6.1, our methodology enhances video LMM alignment through DPO method using rewards from a language model. We elaborate on constructing a video caption dataset in § 6.4. Subsequently, in § 6.4, we discuss the generation of video instruction data and the fine-tuning process of our model. Lastly, § 6.4 details the incorporation of generated captions

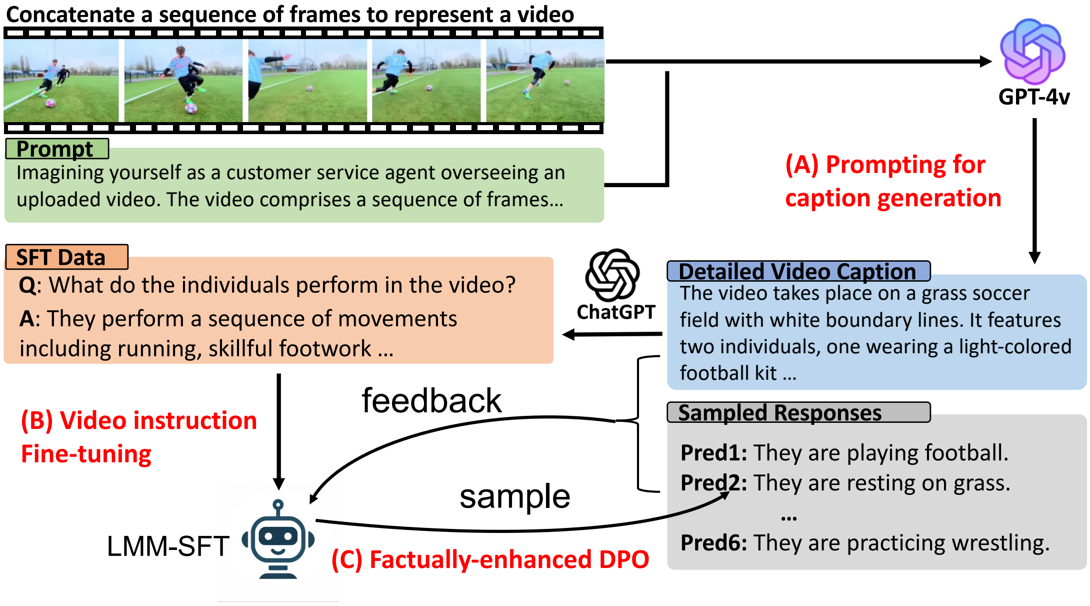
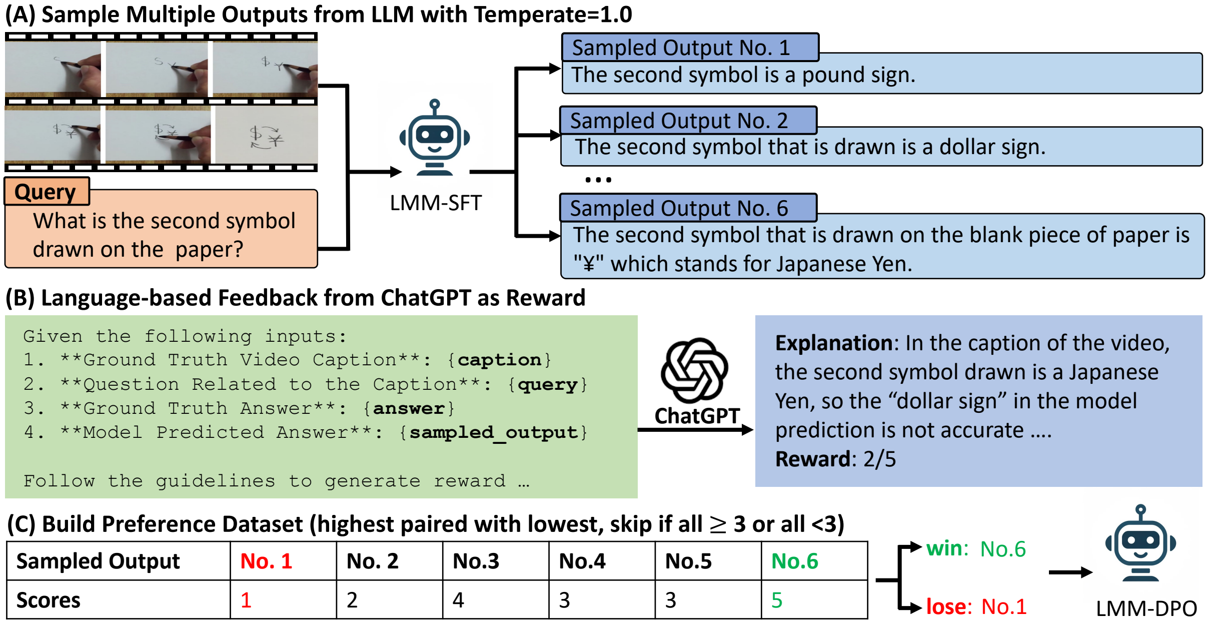

Concatenate a sequence of frames to represent a video
GPT-4v
Prompt
Imagining yourself as a customer service agent overseeing an uploaded video. The video comprises a sequence of frames...
(A) Prompting for caption generation
SFT Data
Q: What do the individuals perform in the video?
A: They perform a sequence of movements including running, skillful footwork ...
ChatGPT
Detailed Video Caption
The video takes place on a grass soccer field with white boundary lines. It features two individuals, one wearing a light-colored football kit ...
(B) Video instruction Fine-tuning
feedback
Sampled Responses
Pred1: They are playing football.
Pred2: They are resting on grass.
...
Pred6: They are practicing wrestling.
sample
LMM-SFT
(C) Factually-enhanced DPO

Figure 6.1: Workflow diagram showing: a) the use of GPT-4V for creating a detailed caption dataset for videos; b) generating video instruction data for SFT; c) integrating captions into a feedback loop for DPO, improving the model's performance on video instruction-following tasks.

as a feedback mechanism for DPO method to refine our model's factual alignment in video instruction-following tasks.

Prompting GPT-4V Model for Detailed Video Caption Distillation The selection of dataset includes videos from three sources: WebVid (400k) and VIDAL (450k) ActivityNet (50k) datasets. WebVid and VIDAL videos are in the general domain sourced from YouTube, and ActivityNet videos focus on human activities. The three datasets together result in a comprehensive collection of 900k videos. To accommodate the requirement that GPT-4V only takes images as input, we preprocess videos by uniformly extracting ten frames per video content. These frames are then concatenated into a sequence to serve as a proxy for the video. We use GPT-4V to generate a coherent caption for the represented video based on the frame sequence. The prompt covers temporal dynamics, world knowledge, object attributes, spatial relationships, aesthetic assessments, etc., with the goal of comprehensively understanding the video contents (examples in fig. 6.8).

SFT with Generated Video Instruction Data from Detailed Caption To generate video instruction-following data for SFT, we adopt a similar methodology outlined in Video-ChatGPT [136]. Specifically, we first randomly sample 300k video captions and then employ ChatGPT to generate 3 question-answer pairs conditioned on each caption. We release the 900k instruction-following


(A) Sample Multiple Outputs from LLM with Temperate=1.0
Query
What is the second symbol drawn on the paper?
LMM-SFT
Sampled Output No. 1
The second symbol is a pound sign.
Sampled Output No. 2
The second symbol that is drawn is a dollar sign.
…
Sampled Output No. 6
The second symbol that is drawn on the blank piece of paper is "¥" which stands for Japanese Yen.
(B) Language-based Feedback from ChatGPT as Reward
ChatGPT
(C) Build Preference Dataset (highest paired with lowest, skip if all ≥ 3 or all <3)
win: No.6
lose: No.1
LMM-DPO

```
Given the following inputs:
1. **Ground Truth Video Caption**: {caption}
2. **Question Related to the Caption**: {query}
3. **Ground Truth Answer**: {answer}
4. **Model Predicted Answer**: {sampled_output}

Follow the guidelines to generate reward …
```

Explanation: In the caption of the video, the second symbol drawn is a Japanese Yen, so the "dollar sign" in the model prediction is not accurate ….
Reward: 2/5

Sampled Output	No. 1	No. 2	No.3	No.4	No.5	No.6
Scores	1	2	4	3	3	5

Figure 6.2: Detailed illustration of the proposed factually-enhanced DPO method.

data to public, but we only use a random subset of 240k for our training. This approach ensures that the instructional data remains factually consistent with the content of the detailed captions.

DPO with Language Model Reward Acquiring high-quality on-policy preference data can be costly and labor-intensive. Although GPT-4V can be used for reward distillation, for video data, its high computation cost[1], slow response, and limited accessibility hinder scalability. We propose a cost-efficient method to generate reward data for DPO using detailed video captions as supporting evidence, as shown in fig. 6.2.

Initially, we randomly select a subset of 20k instruction pairs from the dataset described in § 6.4. The SFT model generates six responses per input at a temperature of 1.0. This procedure results in 120k question-answer pairs. Subsequently, we employ ChatGPT to evaluate the model responses based on the ground truth answer and detailed description. ChatGPT generates an output that includes a natural language explanation as chain-of-thought step, followed by a numerical reward score on a scale from 1 to 5, indicating the overall quality.

For each video and question pair, we randomly select an answer with a score ≥ 3 as positive example, and an answer with a score below 3 as negative. Cases where all responses are uniformly scored above or below 3 are excluded from the dataset. After the selection process, approximately 17k training instances are compiled for DPO training. Formally, the dataset is denoted as

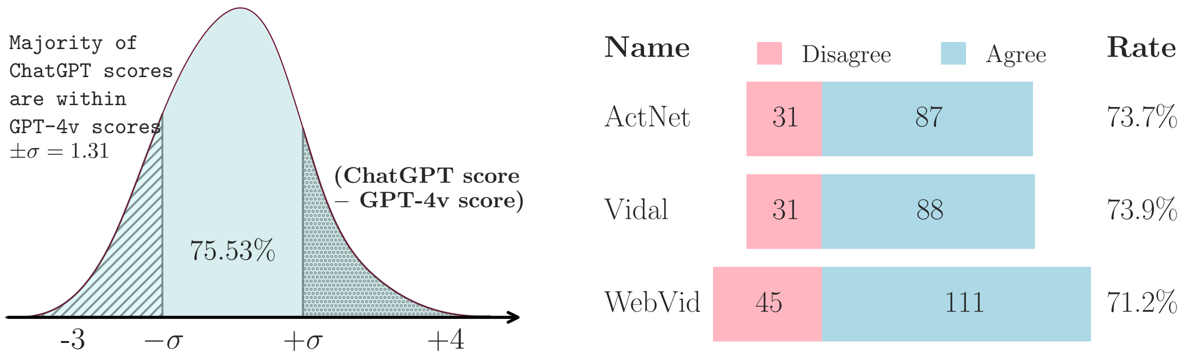
[1] Video representation is typically encoded with 2048 tokens, while our captions only uses roughly 140 tokens.

$\mathcal{D}_{DPO} = \{(\mathcal{V}, x, y_w, y_l)\}$, where $\mathcal{V}$ is the video, x is the question, y_w and y_l are the positive and negative responses. The DPO objective is defined as below:

$$\mathcal{L}_{\text{DPO}}\left(\pi_\theta; \pi_{\text{ref}}\right) = -\mathbb{E}_{(\mathcal{V},x,y_w,y_l)\sim\mathcal{D}_{DPO}}\Bigg[\log\sigma\left(\beta\log\frac{\pi_\theta\left(y_w \mid x, \mathcal{V}\right)}{\pi_{\text{ref}}\left(y_w \mid x, \mathcal{V}\right)} - \beta\log\frac{\pi_\theta\left(y_l \mid x, \mathcal{V}\right)}{\pi_{\text{ref}}\left(y_l \mid x, \mathcal{V}\right)}\right)\Bigg],$$

where π_θ is the policy model to be optimized and π_{ref} is the base reference model, both models are initialized with SFT weights. σ is the logistic function and β is set to 0.1.

For on-policy reward generation, our method incurs a cost of less than $20, under a pricing model of $1.5 per million tokens. In comparison, previous methods of preference data collection, such as in [219], required an expenditure of $3,000 to gather 10k human preference data points. Additionally, the method proposed by [138], which employs GPT-4V for reward data labeling, incurs a significantly higher cost—$30 per million tokens—and demonstrates considerably slower inference speeds.



(a) Score Difference Distribution

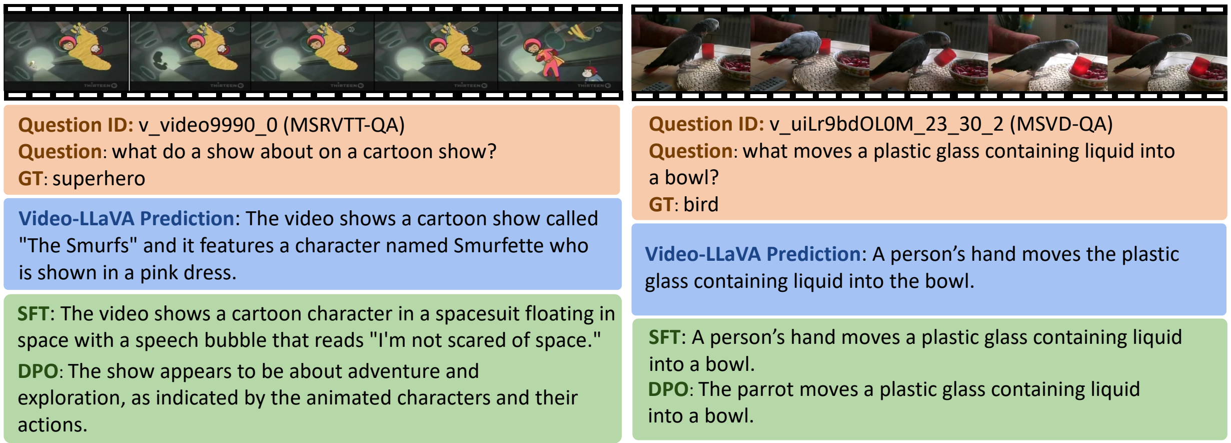
(b) Preference Agreement Rate

Figure 6.3: Assessing Evaluator Quality Using Captions in Place of Frames. (a) The distribution of evaluation score differences between ChatGPT (with caption as proxy) and GPT-4V (directly on frames) evaluations. (b) The rate of preference agreement between ChatGPT and GPT-4V as evaluators.

Assessment of Evaluator with GPT-4V Caption as Video Content To assess the effectiveness of our proposed reward assignment method, we conducted a comparative analysis the GPT-4V used as a video QA evaluator. Our method utilizes detailed captions as a proxy of actual video frames, while GPT-4V directly takes in video frames as inputs. Both reward systems follow the same set of guidelines for scoring reward.

To compare the two methods, we sample 200 videos from each of the WebVid, VIDAL, and ActivityNet datasets, each associated with one question and two model predictions from our SFT model, with one preferred and one dispreferred by ChatGPT. This results in $1,200$ examples, for which we used GPT-4V to assign scores. Filtering through the Azure API backend resulted in 196, 151, and 143 videos from each dataset, respectively, having both answers evaluated. The average scores of all examples from ChatGPT and GPT-4V evaluations were 2.9 and 3.5 respectively, indicating a tendency of GPT-4V to yield slightly positive evaluations. The Pearson Correlation Coefficient (PCC) of 0.47 ($p < 0.01$) suggests a moderate positive correlation. In fig. 6.3 (left), the distribution of the difference between ChatGPT and GPT-4V scores reveals that majority ($> 75\%$) of ChatGPT scores fall within one standard deviation ($\sigma = 1.31$) of GPT-4V scores. Additionally, in fig. 6.3 (right), the agreement on preference between ChatGPT and GPT-4V, excluding ties, exceeded 70%. These findings cautiously support our benchmark's applicability in video QA evaluation. Further refinements for better alignment—such as incorporating Likert scales [285] or GPT-4 evaluation—are areas for future research.

Human Annotation of Captions: To evaluate the quality of the distilled captions, we conducted human annotations focusing on two aspects: coverage and accuracy (hallucination). Annotators were asked to assess each caption by identifying the number of missing items and the number of incorrect facts. The assessment was performed on a sample of 75 videos, with 25 from each domain. The results showed that annotators identified a total of 21 inaccurate items across 14 videos (accuracy: 81%) and 12 missing items across 8 videos (accuracy: 89%). Annotated examples are provided in § 6.6.



Question ID: v_video9990_0 (MSRVTT-QA)
Question: what do a show about on a cartoon show?
GT: superhero

Video-LLaVA Prediction: The video shows a cartoon show called "The Smurfs" and it features a character named Smurfette who is shown in a pink dress.

SFT: The video shows a cartoon character in a spacesuit floating in space with a speech bubble that reads "I'm not scared of space."
DPO: The show appears to be about adventure and exploration, as indicated by the animated characters and their actions.

Question ID: v_uiLr9bdOL0M_23_30_2 (MSVD-QA)
Question: what moves a plastic glass containing liquid into a bowl?
GT: bird

Video-LLaVA Prediction: A person's hand moves the plastic glass containing liquid into the bowl.

SFT: A person's hand moves a plastic glass containing liquid into a bowl.
DPO: The parrot moves a plastic glass containing liquid into a bowl.

Figure 6.4: Examples from MSRVTT-QA and MSVD-QA showcase that our LLaVA-Hound-DPO generates better responses, and reveal key limitations of the existing benchmark evaluation.

Methods	LLM Size	Existing Video QA Benchmark from [163]					
		MSVD-QA		MSRVTT-QA		TGIF-QA	
		Acc.	Score	Acc.	Score	Acc.	Score
FrozenBiLM [264]∗	1B	32.2	-	16.8	-	41.0	-
VideoLLaMA [274]∗	7B	51.6	2.5	29.6	1.8	-	-
LLaMA-Adapter [277]∗	7B	54.9	3.1	43.8	2.7	-	-
VideoChat [136]∗	7B	56.3	2.8	45.0	2.5	34.4	2.3
BT-Adapter [155]∗	7B	67.5	3.7	57.0	3.2	-	-
Video-ChatGPT [163]	7B	68.6	3.8	58.9	3.4	47.8	3.2
Chat-UniVi [105]	7B	70.0	3.8	53.1	3.1	46.1	3.1
VideoChat2 [137]	7B	70.0	3.9	54.1	3.3	-	-
Video-LLaVA [150]	7B	71.8	3.9	59.0	3.4	48.4	3.2
LLaMA-VID [144]	7B	72.6	3.9	58.7	3.4	49.2	3.3
LLaMA-VID [144]	13B	74.3	4.0	59.8	3.4	50.8	3.3
VLM-RLAIF [4]∗	7B	76.4	4.0	63.0	3.4	-	-
LLAVA-HOUND-SFT	7B	75.7	3.9	58.7	3.3	53.5	3.3
LLAVA-HOUND-DPO	7B	**80.7**	**4.1**	**70.2**	**3.7**	**61.4**	**3.5**

Table 6.1: **Evaluation of Model Performance on Zero-Shot Video Question Answering Benchmarks Using gpt-3.5-turbo-0613.** Models denoted with ∗ have their results directly sourced from their original publications. Caution is advised when interpreting these results; see Appendix 6.6 for an in-depth analysis of evaluation challenges. All other baseline models were reproduced by our team.

No.	Methods	Next-QA	
		Acc.	Score
[1]	Video-ChatGPT [163]	45.23	2.09
[2]	LLaMA-VID-7B [144]	49.43	3.24
[4]	Chat-UniVi [105]	47.62	3.14
[5]	Video-LLaVA [150]	48.97	3.25
[6]	LLAVA-HOUND-SFT	60.60	3.51
[7]	LLAVA-HOUND-DPO	**74.27**	**3.74**

Table 6.2: Evaluation on Next-QA benchmark using gpt-3.5-turbo-0611 on official test set.

6.5 Empirical Evaluation

We adopt Video-LLaVA [149] as the backbone of our video LMM, but our method can be applied to any other architectures as well.
Caption Pre-training Stage (LLAVA-HOUND-PT): We use captioning data including 650k image caption data from ALLaVA [36] and our distilled 900k video caption. We freeze the visual encoder and fine-tune the MLP projector and LLM, with learning rate 2e-5 and batch size 128.
SFT Stage (LLAVA-HOUND-SFT): We use 600k image instruction data from ALLaVA and our generated 240k video instruction data, with learning rate 5e-6 and batch size 128.
DPO training Stage (LLAVA-HOUND-DPO): We use the 17k preference data introduced in § 6.4 for DPO training. Following [100], we train our policy model with full model training for 3 epochs with learning rate 5e-7, and a batch size of 128. All the experiments are performed on 8 A100 gpus.

Benchmark Evaluation

Dataset and Testing Environment. We evaluate model performance on four benchmark datasets: MSVD-QA [35], MSRVTT-QA [260], TGIF-QA [102], and Next-QA [256] using ChatGPT with version gpt-3.5-turbo-0611 to assess model predictions. The evaluation prompts follow [163]. In our experiment, we found that different ChatGPT versions have high impact on absolute score of metric, but the overall ranking of models is relatively stable. We select gpt-3.5-turbo-0613 due to its closeness to the reported score in Video-LLaVA paper. Further details on the selection rationale and evaluation pitfalls are discussed in Appendix 6.6.

Baseline Selection. We select video LMM models that have demonstrated SOTA performance with with accessible code and checkpoints at the time of paper writing, specifically including Video-LLaVA, which is also our choice of architecture. We replicate results including Video-ChatGPT [163], LLaMA-VID [144] (7B and 13B), Chat-UniVi [105], and Video-LLaVA [150]. We copy the results from additional baselines including FrozenBiLM [264], VideoChat [136] and VideoLLaMA [274], sourced from their original publication.

Results. In table 6.1, our analysis shows that within the SFT models, LLaMA-VID-7B and Video-LLaVA exhibit comparable performance, with LLaMA-VID-13B performing the best. Our LLAVA-HOUND-SFT model achieves comparable performance to LLaMA-VID-13B. Incorporating preference modeling, LLAVA-HOUND-DPO achieves an average accuracy of 70.75%, surpassing LLAVA-HOUND-SFT, which has an average accuracy of 62.65%, by 8.1%. Furthermore, LLAVA-HOUND-DPO exhibits superior accuracy compared to other RL methods such as

VLM-RLAIF. In table 6.2, our model demonstrated consistent result on a relative new benchmark Next-QA.

Error Analysis. Figure 6.4 illustrates two examples. In the left example, LLaVA-Hound-SFT provides an accurate description of the video's first half but introduces a hallucination with the phrase "I'm not scared of space," absent in the video content. LLaVA-Hound-DPO yields a more accurate inference. In the right example, both LLaVA-Hound-SFT and Video-LLaVA models produce incorrect inferences, whereas LLaVA-Hound-DPO successfully correctly identifies the subject in the video.

Open-ended QA Analysis In this section, we conduct analysis on open-ended long-form QA with a proposed development benchmark. Specifically, we select 2,000 videos from each source: WebVid [17], VIDAL [286], ActivityNet [71], MSRVTT [260], MSVD [35], TGIF [102], and Something-something V2 (SSV2) [79]. For each video, ChatGPT was utilized to generate three QA pairs based on the detailed captions, and we evaluate model predictions with our reward mechanism. WebVid, VIDAL, ActivityNet are classified as in-domain, which are involved in the model's training pipeline. MSRVTT, MSVD, TGIF, SSV2 are classified as out-of-domain.

The evaluation reveals insights into (1) the quality of long-form open-ended QA, (2) in-domain and out-of-domain generalization, and (3) Ablations on SFT and DPO experiments. Additionally, we select our best performing model on the development bench before evaluating on public benchmarks, which avoids tuning hyperparameters on test data. Comparisons are shown in § 6.6.

Domain Generalization: Table 6.3 and table 6.4 shows the in-domain and out-of-domain evaluation. SFT with our data tends to perform better both in- and out-of-domain, and DPO further enhances the model performance, showing the effectiveness of preference modeling.

Video LMM without Video Instruction: [8] in table 6.3 is baseline trained with only image instruction fine-tuned on LLaVA-Hound-PT, which achieves an average accuracy of 65.97%, comparable to the LLaVA-Hound-SFT model's 66.06% in in-domain QA scenarios. However, its performance significantly drops in out-of-domain QA contexts (49.32% vs. 56.50%), suggesting that Video QA training could potentially enhance generalization capabilities.

Quality of Generated SFT: [9] substitutes our generated video QA with the Video-ChatGPT dataset for Video-LLaVA fine-tuning. A comparison between the findings of [9] and [6] reveals a marginal performance disparity of 0.2% in average accuracy, indicating that the quality of our generated QA closely parallels that of the existing video QA datasets. Given the similar quality in SFT data, the large gain of [6] over [5] can be reasonably concluded from large-scale pre-training on video captions.

Unfreeze MLP: The comparison between [10] and [7] reveals a significant decrease in per-

No.	Methods	Proposed Video QA Benchmark (In-domain)					
		ActivityNet-QA		VIDAL-QA		WebVid-QA	
		Acc.	Score	Acc.	Score	Acc.	Score
[1]	Video-ChatGPT [163]	34.17	2.19	29.35	2.10	38.88	2.27
[2]	LLaMA-VID-7B [144]	36.54	2.27	30.58	2.15	36.99	2.24
[3]	LLaMA-VID-13B [144]	37.33	2.29	32.50	2.18	39.73	2.30
[4]	Chat-UniVi [105]	39.35	2.32	31.40	2.16	40.05	2.31
[5]	Video-LLaVA [150]	41.35	2.38	34.30	2.24	42.47	2.39
[6]	LLaVA-Hound-SFT	66.62	3.05	60.50	2.88	71.07	3.17
[7]	LLaVA-Hound-DPO	**76.62**	**3.18**	**70.06**	**3.04**	**79.82**	**3.29**
[8]	LLaVA-Hound-PT + Image Inst.	69.31	3.09	60.57	2.85	68.03	3.02
[9]	LLaVA-Hound-PT + VChat	67.34	3.02	62.33	2.89	68.98	3.00
[10]	LLaVA-Hound-DPO + training MLP	71.89	3.10	65.57	2.95	75.37	3.21
[11]	LLaVA-Hound-SFT + Self-play	64.11	2.85	56.28	2.68	67.89	2.95
[12]	LLaVA-Hound-DPO w/ lr3e-7	71.13	3.08	64.90	2.92	73.25	3.17

Table 6.3: Our proposed video QA benchmark evaluation on in-domain dataset using gpt-3.5-turbo-0301, with detailed captions as supporting evidence.

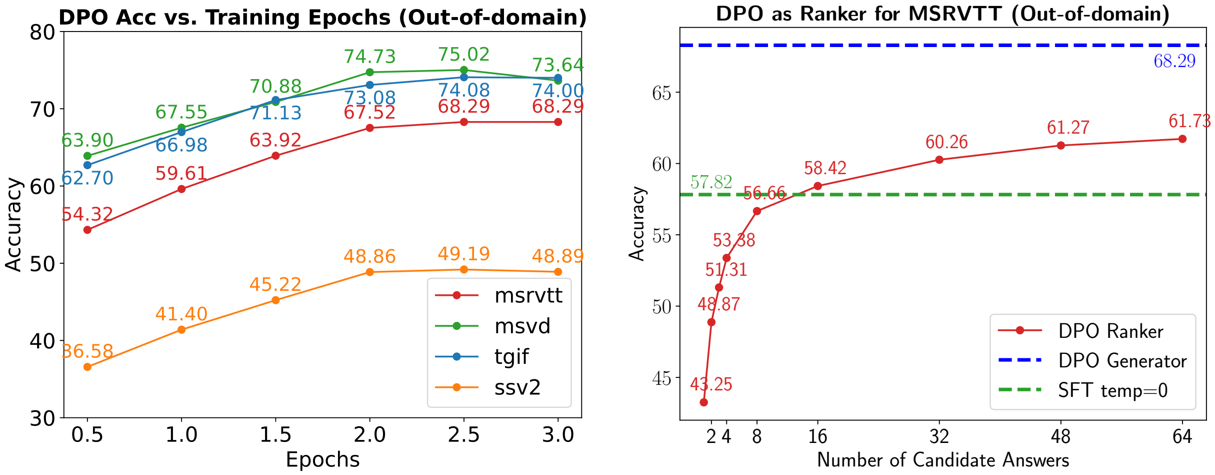
Methods	Proposed Video QA Benchmark (Out-of-domain)							
	MSVD-QA		MSRVTT-QA		TGIF-QA		SSV2-QA	
	Acc.	Score	Acc.	Score	Acc.	Score	Acc.	Score
Video-ChatGPT [163]	34.06	2.20	25.65	1.98	31.35	2.09	19.36	1.75
LLaMA-VID-7B [144]	34.14	2.21	25.02	1.99	27.18	2.00	22.16	1.84
LLaMA-VID-13B [144]	35.81	2.25	26.34	2.02	27.58	2.01	21.98	1.83
Chat-UniVi [105]	35.61	2.23	25.89	2.01	33.23	2.13	20.59	1.79
Video-LLaVA [150]	39.46	2.37	30.78	2.15	32.95	2.18	24.31	1.90
LLaVA-Hound-SFT	66.99	3.09	57.82	2.85	66.13	3.07	35.07	2.23
LLaVA-Hound-DPO	**73.64**	**3.12**	**68.29**	**2.98**	**74.00**	**3.12**	**48.89**	**2.53**
LLaVA-Hound-PT + Image Inst.	65.19	2.96	48.66	2.52	53.83	2.62	29.60	2.04

Table 6.4: Our proposed video QA benchmark evaluation on out-of-domain dataset using gpt-3.5-turbo-0301, with detailed captions as supporting evidence.

formance when the MLP is unfrozen during DPO training. Despite this drop, however, the performance remains superior to that of the SFT baseline.

Smaller Learning Rate: The comparison between [12] and [7] reveals that using a smaller learning rate of 3e-7 (vs. 5e-7) results in a decreasing of model performance. This highlights the future improvements by finding better hyperparameters.

Self-Play vs. DPO: [51] introduced a self-play methodology for DPO training, which designates ground truth answers as preferred and model-generated responses as dispreferred. When comparing the results of [11] with those in [6], a notable decrease in accuracy by 3% from the SFT model is observed, suggesting that self-play may be less effective for video LMM alignment, and introducing reward model is helpful.



(a) DPO Test Set Accuracy

(b) DPO Performance: Generator vs. Ranker

Figure 6.5: DPO Model Performance. (a) The test set accuracy of the DPO model with respect to the number of training epochs. (b) A comparison of DPO model performance as a generator versus a ranker.

DPO Accuracy vs. Training Epochs. The left of fig. 6.5 depicts the generalization performance of the model on out-of-domain video QA tasks with respect to the number of training epochs. We observe a consistent enhancement in model performance among datasets during the initial 0 to 2 epochs, with peak performance materializing at around 2.5 epochs, which corresponds to 350 training steps.

DPO as Ranker vs. Generator. Following [94], we compare the performance of employing the DPO model as a ranker for candidate answers produced by the SFT model, operating at a temperature setting of 1.0. As depicted on the right in fig. 6.5, we illustrate the test accuracy progression through the selection of the best among N candidates by the DPO ranker. Initial observations indicate that the SFT model, when set to a temperature of 1.0, demonstrates a reduced accuracy (43.3%) compared to that achieved through greedy decoding (57.8%). A steady enhancement in performance is noted as the number of candidates increases, plateauing at an accuracy of approximately 62% with 64 candidates. This performance, however, falls short when compared with the direct application of the DPO model for answer generation, which yields an accuracy of 68.29%. This difference suggests the stronger generalization of DPO model in answer generation, despite it is trained on a reward classification loss. The contradictory results to [94] may be due to the difference of tasks, i.e. Math vs. Video QA. Refer to § 6.6 for more results.

6.6 Understanding the Models

Methods	LLM Size	MSVD-QA		MSRVTT-QA		TGIF-QA		Summary	
		Acc.	Score	Acc.	Score	Acc.	Score	Avg Acc.	Rank
				gpt-3.5-turbo-0301 evaluation					
Video-ChatGPT [163]	7B	78.62	4.00	71.67	3.63	56.31	3.45	68.87	6
LLaMA-VID [144]	7B	82.57	4.12	71.94	3.65	59.00	3.63	71.17	4
LLaMA-VID [144]	13B	83.72	4.16	73.63	3.68	59.72	3.66	72.36	3
Chat-UniVi [105]	7B	80.52	4.02	66.92	3.41	57.73	3.49	68.39	7
Video-LLaVA [150]	7B	81.44	4.08	73.29	3.65	58.34	3.61	71.02	5
LLaVA-Hound-SFT	7B	85.65	4.10	73.85	3.62	64.98	3.65	74.83	2
LLaVA-Hound-DPO	7B	**88.50**	**4.20**	**82.10**	**3.84**	**75.48**	**3.81**	**82.03**	1
				gpt-3.5-turbo-0613 evaluation					
Video-ChatGPT [163]	7B	68.55	3.80	58.90	3.36	47.83	3.21	58.43	6
LLaMA-VID [144]	7B	72.62	3.92	58.73	3.38	49.21	3.28	60.19	4
LLaMA-VID [144]	13B	74.29	3.96	59.82	3.41	50.83	3.33	61.65	3
Chat-UniVi [105]	7B	70.01	3.79	53.08	3.14	46.09	3.12	56.39	7
Video-LLaVA [150]	7B	71.75	3.88	58.97	3.39	48.39	3.24	59.70	5
LLaVA-Hound-SFT	7B	75.70	3.86	58.73	3.31	53.51	3.30	62.65	2
LLaVA-Hound-DPO	7B	**80.73**	**4.07**	**70.15**	**3.66**	**61.38**	**3.46**	**70.75**	1
				gpt-3.5-turbo-1106 evaluation					
Video-ChatGPT [163]	7B	73.02	4.01	62.09	3.61	47.76	3.36	60.96	6
LLaMA-VID [144]	7B	75.49	4.08	62.09	3.61	51.72	3.47	63.10	4
LLaMA-VID [144]	13B	76.97	4.10	63.16	3.61	52.53	3.50	64.22	3
Chat-UniVi [105]	7B	72.22	3.92	55.02	3.35	48.16	3.31	58.47	7
Video-LLaVA [150]	7B	74.76	4.04	62.70	3.60	51.21	3.45	62.89	5
LLaVA-Hound-SFT	7B	81.09	4.08	64.13	3.57	58.05	3.53	67.76	2
LLaVA-Hound-DPO	7B	**86.05**	**4.23**	**76.75**	**3.85**	**70.02**	**3.71**	**77.61**	1

Table 6.5: **Performance Evaluation Across ChatGPT Versions on Zero-Shot Video Question Answering Benchmarks.** This table compares the performance of state-of-the-art video LMMs evaluated under different ChatGPT versions. The absolute performance metrics scored by ChatGPT vary by versions. However, the comparative ranking of models under the same ChatGPT version is relatively stable.

Effect of ChatGPT Version on Official Benchmark Evaluation. In Table 6.5, we show impact of using different ChatGPT versions on metric scores within zero-shot video question answering benchmarks. Our analysis reveals significant variations in the absolute scores across ChatGPT versions, but based on the average accuracy metric, the relative ranking of models under the same

ChatGPT version shows consistency.

This comparison underscores a critical issue: many prior studies neglect to specify the ChatGPT version used, potentially leading to inaccurate conclusions during evaluation. We advocate for the explicit designation of the ChatGPT version in future evaluations. Analysis from Table 6.5 indicates that the version gpt-3.5-turbo-0613 aligns most closely with the performance of the Video-LLaVA [149] model, serving as the benchmark for model performance comparison in our study.

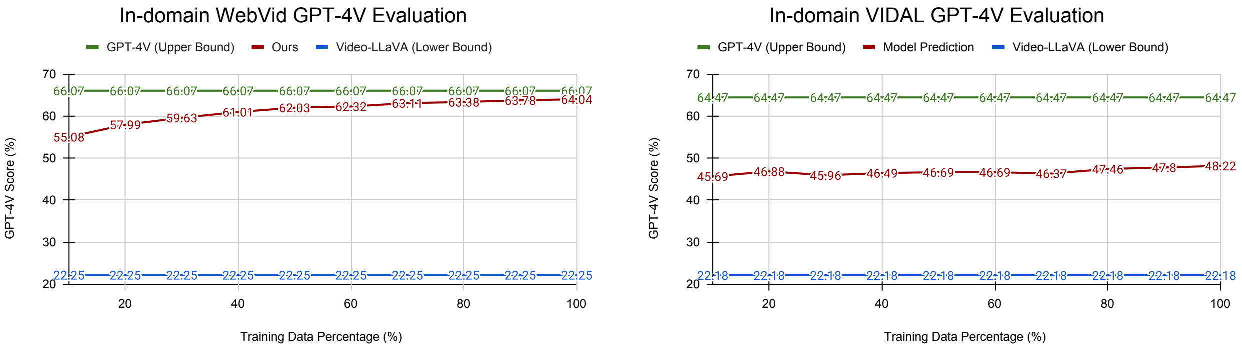


Figure 6.6: Training subsets exhibit varying levels of generalization difficulty. The WebVid subset (left) requires less data compared to the VIDAL subset (right)

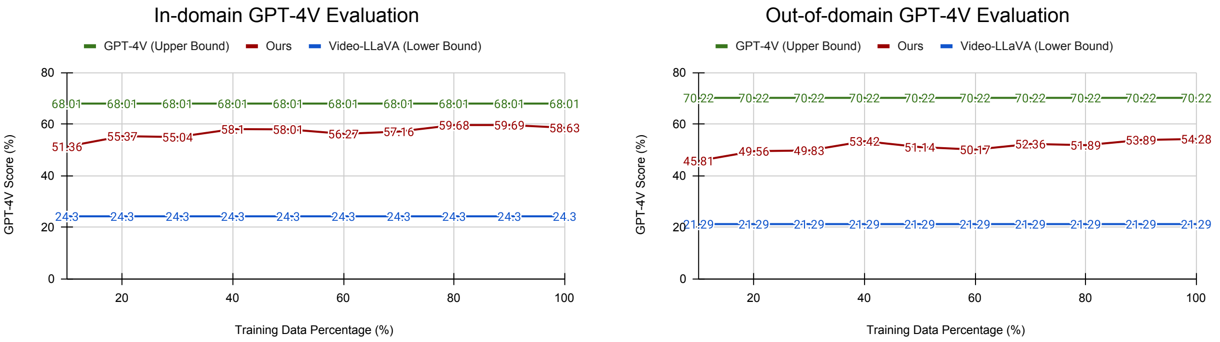
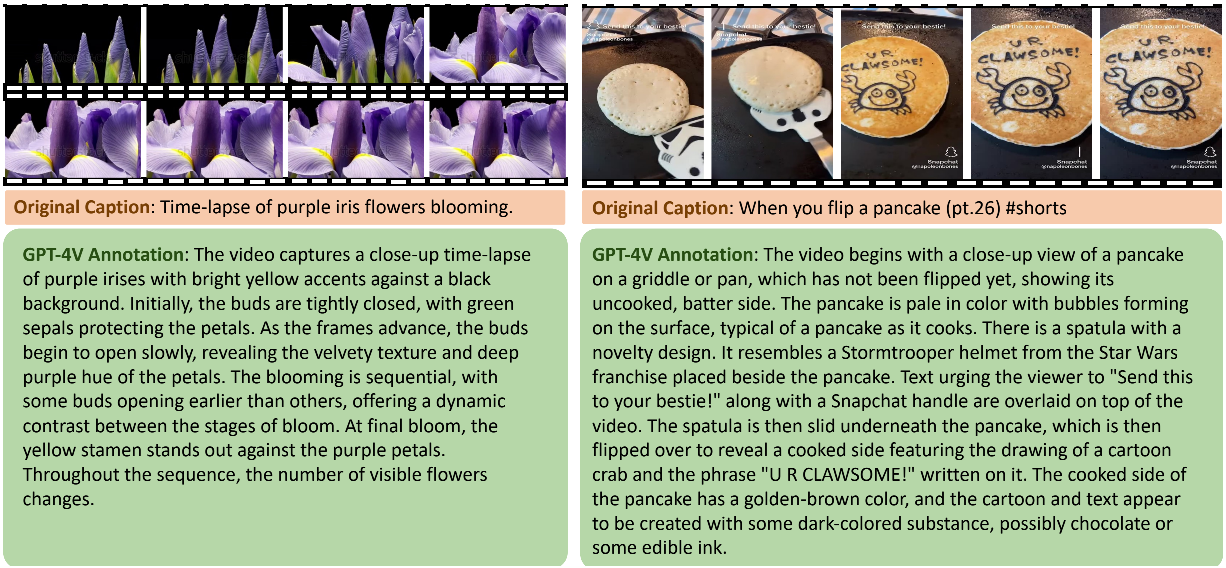


Figure 6.7: The video caption ability w.r.t number of training data evaluated on both in-domain and out-of-domain test videos using GPT-4V.

Evaluation of Captioning Ability from pre-training. In Figure 6.7, we present the video captioning ability of models across various datasets, with a total of 900k distilled data instances. GPT-4V is employed for self-evaluation, serving as the upper-bound performance, while the Video-LLaVA serves for comparative analysis, establishing a baseline. Notably, Video-LLaVA is trained on 54k video QA data instances. However, our first checkpoint, utilizing only 10% of the data, is trained on 90k high-quality caption data instances, likely accounting for the observed performance disparity in the video captioning task. Our results demonstrate that incorporating more distilled data contributes to improved model performance across both in-domain and out-of-domain datasets. Despite these improvements, a performance discrepancy with the GPT-4V model remains. Further, we evaluate the generalization potential in specific data subsets, as shown in fig. 6.6 in the Appendix. These subsets reveal varying degrees of generalization challenges for different types of dataset. For example, the WebVid subset, which concentrates on relatively static scenes, necessitates less data for effective training compared to the VIDAL subset, which is marked by dynamic scene transitions and a diversity of video themes.



Original Caption: Time-lapse of purple iris flowers blooming.

GPT-4V Annotation: The video captures a close-up time-lapse of purple irises with bright yellow accents against a black background. Initially, the buds are tightly closed, with green sepals protecting the petals. As the frames advance, the buds begin to open slowly, revealing the velvety texture and deep purple hue of the petals. The blooming is sequential, with some buds opening earlier than others, offering a dynamic contrast between the stages of bloom. At final bloom, the yellow stamen stands out against the purple petals. Throughout the sequence, the number of visible flowers changes.

Original Caption: When you flip a pancake (pt.26) #shorts

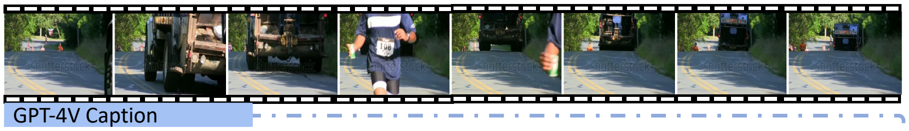
GPT-4V Annotation: The video begins with a close-up view of a pancake on a griddle or pan, which has not been flipped yet, showing its uncooked, batter side. The pancake is pale in color with bubbles forming on the surface, typical of a pancake as it cooks. There is a spatula with a novelty design. It resembles a Stormtrooper helmet from the Star Wars franchise placed beside the pancake. Text urging the viewer to "Send this to your bestie!" along with a Snapchat handle are overlaid on top of the video. The spatula is then slid underneath the pancake, which is then flipped over to reveal a cooked side featuring the drawing of a cartoon crab and the phrase "U R CLAWSOME!" written on it. The cooked side of the pancake has a golden-brown color, and the cartoon and text appear to be created with some dark-colored substance, possibly chocolate or some edible ink.

Figure 6.8: A showcase of our dataset for detailed video caption automatically distilled from GPT-4V via our prompt technique in § 6.4.

GPT-4V Caption Distillation

Human Annotated Examples of Distilled Captions

WebVid Video ID: 2994286



GPT-4V Caption

Initially, the video displays a curved road lined with multiple orange traffic cones and greenery on either side. The focus shifts to a large, dark-colored truck approaching, its engine sound becoming increasingly prominent. The camera angle changes to the rear of the truck, revealing a runner, wearing a blue top with a white number bib marked '108', passing the truck on its left side. The video concludes with the truck's rear-end seen driving away with the traffic cones visible in the background.

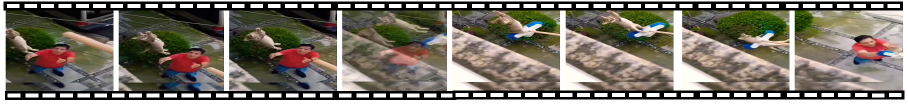
Human Annotation

Inaccuracy (Hallucination):
1. The truck is driving away instead of approaching

Coverage:
1. The runner is holding a cup-like item

Figure 6.9: Human Annotation Analysis: The video frames depict a truck driving away from the camera, but the distilled caption incorrectly describes the direction, which is identified as a hallucination. Additionally, human annotators noted that the GPT-4V model failed to recognize a man holding a cup.

Vidal Video ID: -d3h9kQAMvw



GPT-4V Caption

The cat is observed suspended by a limb caught in an electrical cable. It is a light-colored feline, possibly tan or cream, with a noticeably fluffy tail. Below, the person stands on what appears to be a blue stool or small ladder, reaching out towards the cat in an attempt to free it. The environment includes a green, neatly trimmed bush directly under the cat, and to the side, there appears a section of a vehicle – likely a car, suggesting the proximity to a driveway or parking spot. The pavement where these activities are occurring has a distinct pattern, likely tiles, surrounded by green grass, which borders a residential building. Above, an electrical pole is partially visible with thick cables extending from it – one of which the cat is caught in.

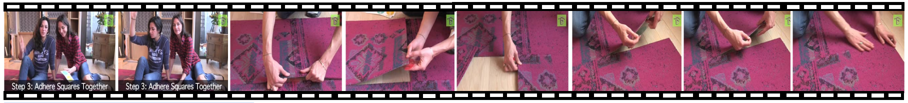
Human Annotation

Inaccuracy (Hallucination):
1.No blue stool or ladder visible

Coverage:
-

Figure 6.10: Human Annotation Analysis: GPT-4V incorrectly identified a man wearing blue pants as a man standing on a blue ladder, which is labeled as a hallucination.

ActivityNet Video ID: v__Y4czVolxvE-Scene-006


Step 3: Adhere Squares Together
Step 3: Adhere Squares Together

GPT-4V Caption

Initially, two people are seen sitting on the floor with crafting materials. They both wear casual clothing: one in a dark blue long-sleeve top with "Johnny's BBQ" written on it and the other in a red and black checkered shirt. They appear to be in a cozy room with a wooden floor and a heater in the background. The next scenes showcase the crafting process in detail, focusing on hands carefully placing square-shaped burgundy materials with patterned textures onto a lighter brown base. The precision of their movements suggests they are experienced in this type of work.

Human Annotation

Inaccuracy (Hallucination):
1. The text on shirt should be “Jimmy's BBQ”

Coverage:
1. Missing text: “Step 3: Adhere Squares Together”

Figure 6.11: Human Annotation Analysis: The example demonstrates OCR hallucination and a coverage issue. The text on the woman's t-shirt is incorrectly labeled as ‘Johnny's BBQ’ instead of ‘Jimmy's BBQ.’ Additionally, the text ‘Step 3: Adhere Squares Together’ is missing from the description.

ActivityNet Video ID: v__wITx73-BXw-Scene-003

GPT-4V Caption

The video showcases multiple scenes from a soccer match being played on a field surrounded by an athletic track. It appears to be in the evening, and a scoreboard is visible in the background indicating the game is underway. The players, wearing contrasting uniforms, are seen running, maneuvering the ball, and aiming at the goal. The crowd can be faintly seen on the sidelines watching the game.

Human Annotation

Inaccuracy (Hallucination):
-

Coverage:
1. “Gopher” and “Coca Cola” on the board
2. Score board shows time left

Figure 6.12: Human Annotation Analysis: The caption does not contain any hallucinations, but some text recognized by human annotators is missing, such as ‘Coca Cola’ and ‘Gopher’ on the scoreboard, as well as the time of the score match shown.

ActivityNet Video ID: v__Wag6CT_0j8-Scene-003

GPT-4V Caption

Throughout the video, there are multiple sequences featuring women in athletic attire engaged in a field hockey match. The players are distributed across a well-maintained, green artificial turf field. Two teams are visible: one in dark blue and white striped jerseys, and the other in white jerseys with black numerals. The white-jersey team appears to be on defense, while the dark-jersey team is in an offensive posture, advancing the ball. Some of the key visual elements include the athletic gear, such as shin guards, field hockey sticks, and cleats that the players are wearing. The background includes features like team benches, spectator stands, and a scoreboard that shows "MONMOUTH 0 2nd RIDER 2". Throughout the various frames of the video, players can be seen running, likely maneuvering the ball, and repositioning to either defend or continue their offensive push. Ambient shadows suggest sunny weather conditions during the recording of this match.

Human Annotation

Inaccuracy (Hallucination):
-

Coverage:
1. Score transition from "MONMOUTH 0 2nd RIDER 2” to "MONMOUTH 1 2nd RIDER 2".

Figure 6.13: Human Annotation Analysis: The figures illustrate a scoring transition that is not recognized by the GPT-4V model, highlighting the challenge of using static frames for accurate captioning.

Video QA Dataset Demonstration To build the test dataset, we curated a dataset comprising 2,000 videos from each source: WebVid [17], VIDAL [286], ActivityNet [71], MSRVTT [260], MSVD [35], TGIF [102], and Something-something V2 (SSV2) [79]. For each video, ChatGPT was utilized to generate three QA pairs based on the detailed captions. The first three datasets (WebVid, VIDAL, ActivityNet) are classified as in-domain, since the captions and QA pairs derived from these sources are used in the model’s training pipeline. Conversely, the remaining datasets (MSRVTT, MSVD, TGIF, SSV2) are classified as out-of-domain, evaluating model’s zero-shot QA ability.

§ 6.6 compares our development benchmark with existing benchmark dataset, we identify several issues with the existing evaluation methods: (1) the auto-generated questions from current benchmarks may be grammatically incorrect or nonsensical, and (2) the answers are limited to a *single* word, which is inadequate for evaluating LMMs in the context of long-form QA. We conduct further analysis on open-ended long-form QA with a proposed development benchmark.

We apply our reward system as described in § 6.4 and report scores from ChatGPT. A score of ≥ 3 is considered correct for accuracy calculations. The development benchmark reveals insights into (1) the quality of long-form open-ended QA, and (2) in-domain and out-of-domain generalization. Additionally, our development benchmark results correlate with existing benchmarks. We recommend that models be evaluated on the development benchmark first, followed by human evaluation.

MSRVTT Video ID: video7012

Existing Video QA Dataset

Q: what is a suit man doing?
A: talk

Q: what does a man seated in a chair discuss?
A: health

Q: who is speaking about the relevance of disorders?
A: man

Our Video QA Dataset

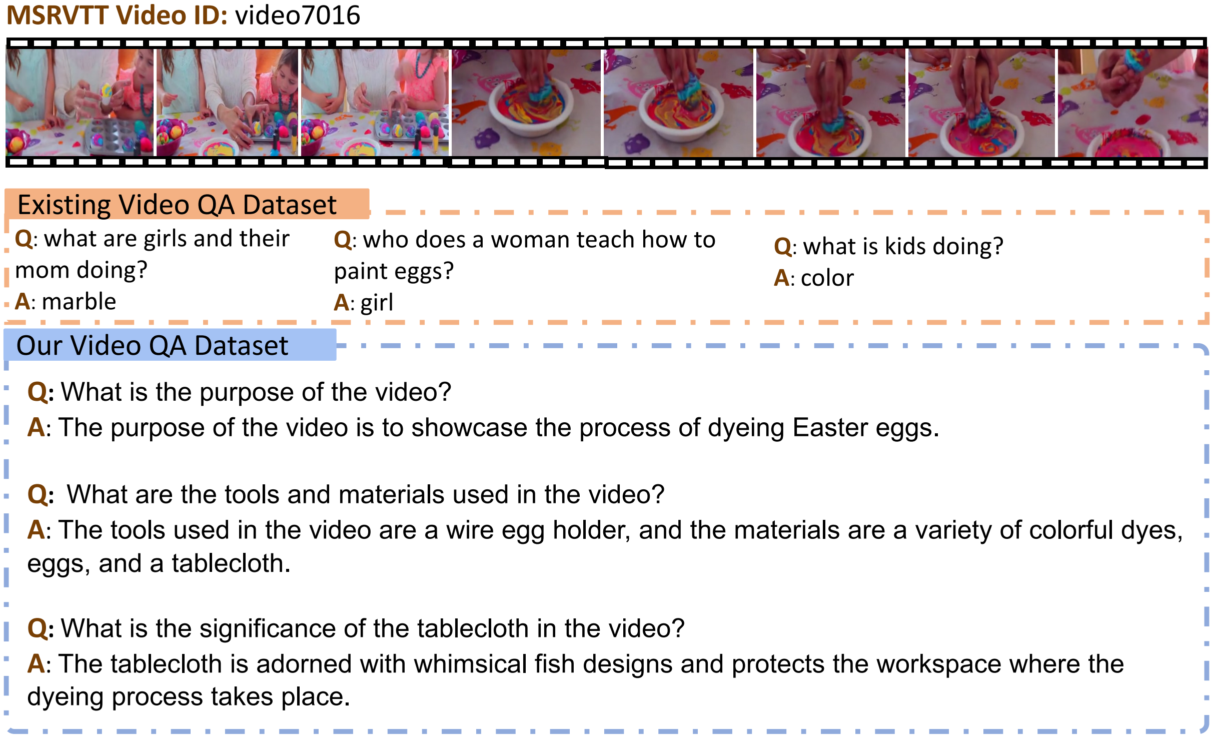
Q: What is the person in the video doing and what does their appearance suggest?

A: The person in the video is sitting in a leather chair against a wooden backdrop with framed art and a small light fixture on the wall and appears to be delivering a speech or presentation. Their appearance suggests that they are confident and engaged in the presentation or interaction.

Q: What decorative items are on the desk in the video and what does the lighting in the room look like?

A: The desk in the video features a couple of decorative items including a blue-striped spherical object and a small chrome lamp with a white shade. The lighting in the room is soft and comes from the person's left side, casting soft shadows.

Figure 6.14: Comparing testing QA in existing benchmark with that in our proposed new benchmark.



MSRVTT Video ID: video7016

Existing Video QA Dataset

Q: what are girls and their mom doing?
A: marble

Q: who does a woman teach how to paint eggs?
A: girl

Q: what is kids doing?
A: color

Our Video QA Dataset

Q: What is the purpose of the video?
A: The purpose of the video is to showcase the process of dyeing Easter eggs.

Q: What are the tools and materials used in the video?
A: The tools used in the video are a wire egg holder, and the materials are a variety of colorful dyes, eggs, and a tablecloth.

Q: What is the significance of the tablecloth in the video?
A: The tablecloth is adorned with whimsical fish designs and protects the workspace where the dyeing process takes place.

Figure 6.15: Comparing testing QA in existing benchmark with that in our proposed new benchmark, example 2.

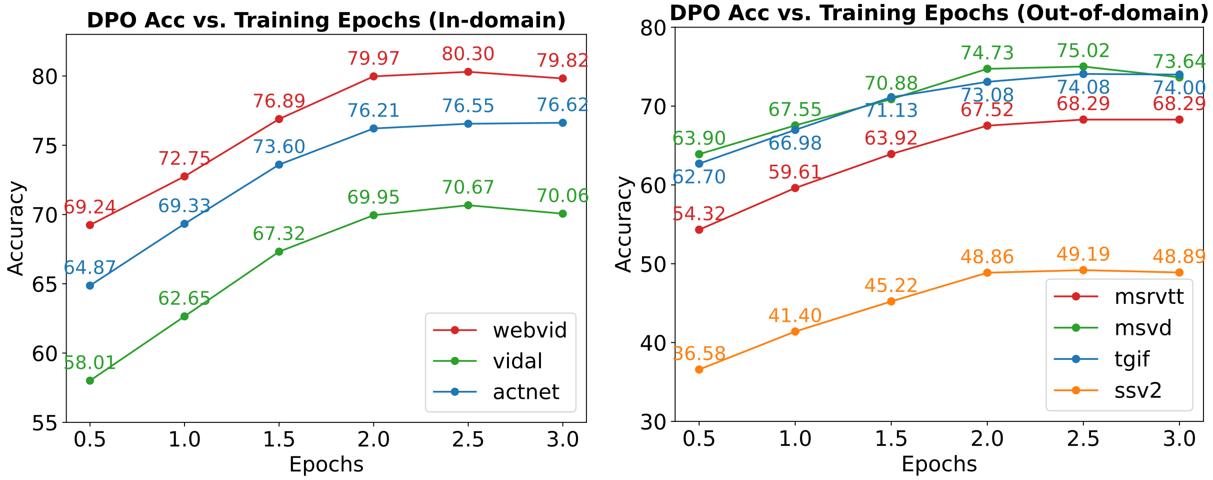


Figure 6.16: Test Set Accuracy of the DPO Model vs. Training Epochs. The figure illustrates a consistent trend in both in-domain and out-of-domain video QA, with peak performance occurring at approximately epoch 2.5, equivalent to 350 training steps.

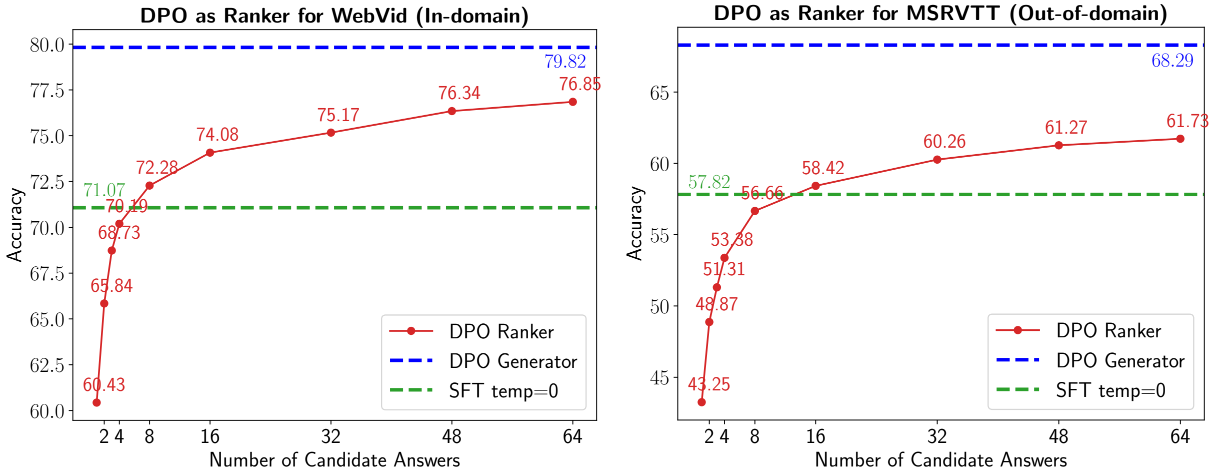


Figure 6.17: Comparison of DPO Model Performance: Ranker vs. Generator. The DPO model serves as a ranker, assigning reward scores to candidate answers generated by the SFT model with a temperature setting of 1.0. Employing the DPO model directly for answer generation results in superior performance compared to its use as a ranker.

Additional DPO Results

6.7 Summary

We study the techniques for effective video LMM alignment. Specifically, we propose an cost-effective reward system that utilizes detailed captions as proxies for video content. We have shown the reward scores is well-aligned with the evaluation metrics of GPT-4V, and DPO training greatly enhances model performance. In addition, we have released 900k detailed video caption, 900k video instruction-following data, and 17k preference data pairs, with a complete code pipeline including pre-training for video captioning, fine-tuning for video instruction following and reinforcement learning with DPO for better LMM alignment.

Chapter 7

Conclusion and Future Directions

This thesis explores multiple strategies to achieve data-efficient multimodal learning, with a focus on reducing the reliance on large labeled datasets while maintaining robust performance across vision and language tasks. At the heart of this thesis is the hypothesis that, with appropriate reasoning capabilities, models can learn robust and generalizable representations by leveraging structured human priors, human-generated weak supervision and common sense knowledge embedded in large-scale pre-trained models.

The first part of this thesis presents methods for incorporating structured human priors into model design and training to improve generalization in low-data settings. By embedding inductive biases drawn from human cognition, our proposed approaches achieve significant improvements in tasks such as handwritten Arabic text recognition and few-shot image recognition. For example, in Chapter 2, we introduce a method for handwritten Arabic text recognition that leverages morphological priors of the Arabic language, allowing accurate recognition under limited supervision. In Chapter 3, we propose a prior-guided data augmentation that generates informative synthetic samples, thereby significantly improving performance in few-shot classification scenarios. These findings in first part demonstrate the effectiveness of integrating structured human priors into model design and training, producing robust results even when annotated data are limited.

The second part of this thesis further reduces the reliance on explicit domain knowledge by leveraging human-generated weak supervision. The focus is primarily on naturally occurring image-text pairs (*e.g.*, alt-text, noisy web captions) or external knowledge sources. In Chapter 4, we demonstrate that visual representations learned from weakly aligned images and text data can match or outperform those trained on large manually labeled datasets, while requiring far less curated data. By aligning images with descriptive text information, the proposed model learns high-level semantic concepts and relationships. Such visual representations benefit from the sematic richness of natural language, which leads to more generalizable and transferable open-world visual understanding. Furthermore, in Chapter 5, we extend this paradigm by integrating image-text pairs extracted from external knowledge bases (*e.g.*, Wikidata) and common sense knowledge based on LLM. This integration enables the model to answer questions that require reasoning beyond visual content. Overall, this part transitions from specialized human priors to broadly distributed weak supervision, which minimizes the need for manual annotation. We demonstrate that large-scale labeled datasets are not the only path to high-quality multimodal representations. Carefully using human-generated weak supervision, such as image captions or alt text, can help models learn more efficiently and improve their ability to reason.

Weak human-generated supervision can introduce noise, which can limit its effectiveness in multimodal learning. This limitation is particularly evident in video understanding tasks, where alt-text or hashtags often correspond to only a single frame or a high-level action. The third part of this thesis addresses this challenge by proposing a preference learning framework that aligns model outputs with evaluations derived from large language models (LLMs). It further extends the potential for human-generated weak supervision by using it to guide model training through preference-based learning. In Chapter 6, we use pre-trained LLMs as evaluators to supervise the training of video understanding models. The model outputs are scored by an LLM-informed reward model and the training is optimized to maximize these preference scores. This method demonstrates that LLM-based feedback can significantly improve the quality and relevance of generated output without requiring manual evaluation for every sample. In this part, we show that LLM-informed feedback can effectively guide multimodal learning in complex video tasks, where fine-grained annotations are difficult to collect and scale.

Across these three parts, this thesis provides a unified perspective on data-efficient multimodal learning by progressively broadening the sources of supervision, from structured human priors to weakly aligned human-generated signals, and finally to LLM-informed feedback. The findings demonstrate that multimodal learning can be achieved without extensive reliance on large-scale labeled datasets. It is possible to substantially reduce the need for manual annotation by strategically leveraging human knowledge, weak supervision, and feedback from pre-trained language models. Extensive experiments highlight the feasibility of learning generalizable, robust multimodal representations across diverse tasks. In general, our thesis offers a practical and scalable foundation for the advancement of multimodal systems in both research and real-world applications.

7.1 Discussions

This thesis builds on advances in language models. We leverage (large) language models to reason across modalities and to reduce the amount of supervised data. However, the information asymmetry persists between language and vision. Vision-to-language mapping has an information bottleneck that loses fine-grained details and collapses ambiguous descriptors (*e.g.*, big, near). Conversely, language-to-vision mapping is underspecified, many images can satisfy a single textual description, and crucial physical or spatial cues may be absent. These factors hinder the efficiency of the sample and the effectiveness of the learning. Although LLMs can improve cross-modal alignment by scalable preference learning, this modality-intrinsic asymmetry cannot be fully eliminated. Evaluation adds another complication. Current open domain assessments are based on LLM-based judges. Different LLM versions, trained with different post-training strategies, exhibit varying preferences and sensitivities to prompt format and response length, which undermines fairness and comparability.

While language provides structured labels and abstractions, images and videos encode dense, fine-grained spatial, temporal, and physical information. Vision-only pretraining is prone to overfitting to low-level cues and fails to induce invariances, object representations, and relational structure. In this thesis, it remains unexplored how to reliably elicit the knowledge embedded in fully pre-trained vision encoders. In Chapter 4, we show that fully trained vision models can be aligned with language through lightweight adapters. However, evaluation remains nontrivial: relying exclusively on linguistic outputs to assess visual quality introduces a language bottleneck that potentially degrades what the visual backbone has learned.

We further propose a vision-language feedback loop to advance data-efficient multimodal learning. However, because this training is largely observational, the model has limited access to causal structure and counterfactual reasoning (*e.g.*, what would happen under alternative actions), which are essential for discovery and innovation. The current learning pipeline lacks several key inductive biases, including object-centric 3D structure, causal dynamics, and hierarchical temporal organization. Humans rarely provide explicit instruction about physical laws (*e.g.*, gravity) or object consistency between viewpoints. Models trained primarily on weak textual supervision often lack deep understanding of underlying mechanisms. As of 2025, state-of-the-art video generators still produce implausible ballet sequences (*e.g.*, extra limbs), errors that can be easily detected by humans. Injecting commonsense knowledge and physical priors implicitly, without sacrificing generalization, is therefore central to fully exploiting high-fidelity data.

At the same time, our proposed framework demonstrates strong breadth by composing and rephrasing knowledge, enabling it to address a wide range of queries. However, learning causal structure at the level of objects and events, and abstracting knowledge from long-context sequence and everyday routines, remains an open challenge that this thesis does not fully address and requires further exploration in future work.

7.2 Future Directions

Although this thesis makes progress toward scalable and data-efficient multimodal learning, there remain several avenues for further research. In particular, we highlight three important directions that could extend and build on the present work.

Visual Representation Learning. The recent success of large language models has revolutionized the field of natural language processing and boosted progress in multimodal research. However, most current systems still rely on CLIP-style contrastive training [152, 282] or combine several separate image encoders [145, 228]. These approaches each capture only part of the visual signal and there is currently no unified method for integrating them into a single visual encoder. Scaling up the model size, such as ViT-22B [59], has led to only marginal improvements, indicating that scaling alone is insufficient without more generalizable forms of supervision. In chapter 6, we propose using human preference signals as an additional training objective. Early results demonstrate that a preference learning framework can help align vision and language objectives and improve generalization to new tasks. Future work should explore how preference-based supervision can provide a more comprehensive and less biased learning signal, aiming for stronger and more generalizable multimodal representations.

Unified Visual Generation and Understanding. Another important direction is the development of unified multimodal models that can jointly perform perception and generation tasks across multiple modalities. Currently, visual understanding tasks (*e.g.*, recognition or question answering) and visual generation tasks (*e.g.*, image or video generation) are addressed by separate models. A unified framework could enable a single model to interpret inputs and generate high-quality outputs across modalities, improving overall efficiency and generalization. In Chapter 4, we propose a joint prediction framework that learns to predict masked image patches and language tokens simultaneously, and demonstrate improved visual understanding abilities with significantly fewer training samples. However, due to computational constraints, we were unable to scale this approach to larger models or datasets. Recent works, such as Transfusion [284] and Emu3 [240], explore unifying vision and language within a single model. Although these methods demonstrate encouraging improvements, they tend to underperform compared to specialized models. Furthermore, there remains an open debate on the use of diffusion-based generation versus next-token prediction, and no conclusive strategy has been reached on how to model different modalities in a unified architecture. More recently, GPT-4o has demonstrated strong performance in instruction following capabilities in image generation tasks, with early evidence suggesting that these gains come from a unified model architecture. Google's Veo3 has pushed this further by simultaneously generating both video and audio. These developments highlight the potential of unified models and motivate future work in the building of instruction follow-up systems capable of understanding and generating multimodal content within a single architecture.

On-Device Personalization and Privacy. A third direction is the development of multimodal learning systems that can personalize and adapt to individual users with limited computational resources and strict privacy constraints. This direction is particularly challenging, as recent advances have focused on scaling up model sizes, making them difficult to deploy on resource-constrained devices. This thesis has taken initial steps toward reducing the data requirements for multimodal learning. Building on this foundation, our aim is to develop systems capable of effective personalization using a limited amount of user data. To address these challenges, future research should focus on lightweight adaptation techniques, which update only a small subset of model parameters and operate without transmitting user data to external servers. Recent advances in hardware-optimized models for mobile platforms [1, 231] further support the feasibility of running multimodal systems locally on user devices. By combining efficient adaption strategies with lightweight architectures, multimodal systems could enable truly user-centric AI: models that not only perform well on standard benchmarks but also continuously adapt to users' preferences, while ensuring privacy and low latency.

Bibliography

[1] Marah Abdin, Jyoti Aneja, Hany Awadalla, Ahmed Awadallah, Ammar Ahmad Awan, Nguyen Bach, Amit Bahree, Arash Bakhtiari, Jianmin Bao, Harkirat Behl, et al. Phi-3 technical report: A highly capable language model locally on your phone. *arXiv preprint arXiv:2404.14219*, 2024. 103

[2] Harsh Agrawal, Karan Desai, Yufei Wang, Xinlei Chen, Rishabh Jain, Mark Johnson, Dhruv Batra, Devi Parikh, Stefan Lee, and Peter Anderson. nocaps: novel object captioning at scale. In *ICCV*, 2019. 51

[3] Angeline Aguinaldo, Ping-Yeh Chiang, Alex Gain, Ameya Patil, Kolten Pearson, and Soheil Feizi. Compressing GANs using knowledge distillation. *arXiv preprint arXiv:1902.00159*, 2019. 23

[4] Daechul Ahn, Yura Choi, Youngjae Yu, Dongyeop Kang, and Jonghyun Choi. Tuning large multimodal models for videos using reinforcement learning from ai feedback. *arXiv preprint arXiv:2402.03746*, 2024. 78, 79, 84

[5] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katie Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *arXiv preprint arXiv:2204.14198*, 2022. 2

[6] Kelsey R Allen, Evan Shelhamer, Hanul Shin, and Joshua B Tenenbaum. Infinite mixture prototypes for few-shot learning. In *ICML*, 2019. 23

[7] Jon Almazán, Albert Gordo, Alicia Fornés, and Ernest Valveny. Word spotting and recognition with embedded attributes. *TPAMI*, 36(12):2552–2566, 2014. 15, 17

[8] Rohan Anil, Gabriel Pereyra, Alexandre Passos, Róbert Ormándi, George E. Dahl, and Geoffrey E. Hinton. Large scale distributed neural network training through online distillation. In *ICLR*, 2018. 23

[9] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. Vqa: Visual question answering. In *CVPR*, 2015. 1, 67

[10] Anthreas Antoniou, Amos Storkey, and Harrison Edwards. Data augmentation generative adversarial networks. In *ICLR*, 2017. 23

[11] Antreas Antoniou and Amos J. Storkey. Learning to learn via self-critique. In *NeurIPS*, 2019. 23

[12] Antreas Antoniou, Harrison Edwards, and Amos Storkey. How to train your MAML. In *ICLR*, 2019. 23

[13] Sören Auer, Christian Bizer, Georgi Kobilarov, Jens Lehmann, Richard Cyganiak, and Zachary Ives. Dbpedia: A nucleus for a web of open data. In *The semantic web*, pages 722–735. Springer, 2007. 60

[14] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *ICLR*, 2015. 10

[15] Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen-vl: A frontier large vision-language model with versatile abilities. *arXiv preprint arXiv:2308.12966*, 2023. 79

[16] Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, et al. Constitutional ai: Harmlessness from ai feedback. *arXiv preprint arXiv:2212.08073*, 2022. 78

[17] Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman. Frozen in time: A joint video and image encoder for end-to-end retrieval. In *ICCV*, 2021. 86, 94

[18] Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman. Frozen in time: A joint video and image encoder for end-to-end retrieval. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1728–1738, 2021. 79

[19] Hangbo Bao, Li Dong, and Furu Wei. Beit: Bert pre-training of image transformers. *ICLR*, 2022. 40, 50, 51

[20] Yoshua Bengio, Samy Bengio, and Jocelyn Cloutier. *Learning a synaptic learning rule*. Université de Montréal, Département d'informatique et de recherche opérationnelle, 1990. 20

[21] Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *ICML*, 2009. 21

[22] Luca Bertinetto, João F. Henriques, Philip H. S. Torr, and Andrea Vedaldi. Meta-learning with differentiable closed-form solvers. In *ICLR*, 2018. 23

[23] Théodore Bluche, Hermann Ney, and Christopher Kermorvant. Feature extraction with

convolutional neural networks for handwritten word recognition. In *ICDAR*, 2013. 8

[24] Theodore Bluche, Hermann Ney, and Christopher Kermorvant. Tandem hmm with convolutional neural network for handwritten word recognition. In *ICASSP*, 2013. 15, 17

[25] Théodore Bluche, Hermann Ney, and Christopher Kermorvant. A comparison of sequence-trained deep neural networks and recurrent neural networks optical modeling for handwriting recognition. In *SLSP*, 2014. 17

[26] Théodore Bluche, Jérôome Louradour, and Ronaldo Messina. Scan, attend and read: End-to-end handwritten paragraph recognition with mdlstm attention. In *NeurIPS*, 2016. 8, 10, 17, 18

[27] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020. 52, 65

[28] Cristian Buciluǎ, Rich Caruana, and Alexandru Niculescu-Mizil. Model compression. In *KDD*, 2006. 23

[29] Michal Bušta, Lukáš Neumann, and Jirı Matas. Deep textspotter: An end-to-end trainable scene text localization and recognition framework. In *ICCV*, 2017. 8

[30] Qingxing Cao, Liang Lin, Yukai Shi, Xiaodan Liang, and Guanbin Li. Attention-aware face hallucination via deep reinforcement learning. In *CVPR*, 2017. 10

[31] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *CVPR*, 2021. 50

[32] William Chan, Navdeep Jaitly, Quoc Le, and Oriol Vinyals. Listen, attend and spell: A neural network for large vocabulary conversational speech recognition. In *ICASSP*, 2016. 10

[33] Wei-Lun Chao, Han-Jia Ye, De-Chuan Zhan, Mark Campbell, and Kilian Q Weinberger. Revisiting meta-learning as supervised learning. *arXiv preprint arXiv:2002.00573*, 2020. 21

[34] Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. Reading wikipedia to answer open-domain questions. *arXiv preprint arXiv:1704.00051*, 2017. 62, 67

[35] David Chen and William B Dolan. Collecting highly parallel data for paraphrase evaluation. In *Proceedings of the 49th annual meeting of the association for computational linguistics: human language technologies*, pages 190–200, 2011. 85, 86, 94

[36] Guiming Hardy Chen, Shunian Chen, Ruifei Zhang, Junying Chen, Xiangbo Wu, Zhiyi Zhang, Zhihong Chen, Jianquan Li, Xiang Wan, and Benyou Wang. Allava: Harnessing gpt4v-synthesized data for a lite vision-language model. *arXiv preprint arXiv:2402.11684*, 2024. 85

[37] Keqin Chen, Zhao Zhang, Weili Zeng, Richong Zhang, Feng Zhu, and Rui Zhao. Shikra: Unleashing multimodal llm's referential dialogue magic. *arXiv preprint arXiv:2306.15195*, 2023. 79

[38] Kezhen Chen, Qiuyuan Huang, Paul Smolensky, Kenneth Forbus, and Jianfeng Gao. Learning inference rules with neural tp-reasoner. In *NeurIPS, workshop*, 2020. 62

[39] Kezhen Chen, Qiuyuan Huang, Yonatan Bisk, Daniel McDuff, and Jianfeng Gao. Kb-vlp: Knowledge based vision and language pretraining. In *ICML, workshop*, 2021. 62

[40] Kezhen Chen, Qiuyuan Huang, Daniel McDuff, Xiang Gao, Hamid Palangi, Jianfeng Wang, Kenneth Forbus, and Jianfeng Gao. Nice: Neural image commenting with empathy. In *EMNLP*, 2021. 61

[41] Lin Chen, Xilin Wei, Jinsong Li, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Zehui Chen, Haodong Duan, Zhenyu Tang, Li Yuan, et al. Sharegpt4video: Improving video understanding and generation with better captions. *NeurIPS*, 2024. 79

[42] Mark Chen, Alec Radford, Rewon Child, Jeffrey Wu, Heewoo Jun, David Luan, and Ilya Sutskever. Generative pretraining from pixels. In *ICML*, 2020. 40

[43] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, 2020. 22, 24

[44] Wei-Yu Chen, Yen-Cheng Liu, Zsolt Kira, Yu-Chiang Frank Wang, and Jia-Bin Huang. A closer look at few-shot classification. In *ICLR*, 2019. 23, 25, 28, 29, 31, 32, 33

[45] Xi Chen, Xiao Wang, Soravit Changpinyo, AJ Piergiovanni, Piotr Padlewski, Daniel Salz, Sebastian Goodman, Adam Grycner, Basil Mustafa, Lucas Beyer, et al. Pali: A jointly-scaled multilingual language-image model. *arXiv preprint arXiv:2209.06794*, 2022. 2

[46] Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. In *CVPR*, 2021. 24

[47] Xinlei Chen, Saining Xie, and Kaiming He. An empirical study of training self-supervised vision transformers. In *ICCV*, 2021. 50

[48] Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. Uniter: Universal image-text representation learning. *Proceedings of ECCV*, 2019. 1

[49] Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. Uniter: Universal image-text representation learning. In *ECCV*, 2020. 1, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47

[50] Zitian Chen, Yanwei Fu, Yu-Xiong Wang, Lin Ma, Wei Liu, and Martial Hebert. Image deformation meta-networks for one-shot learning. In *CVPR*, 2019. 23

[51] Zixiang Chen, Yihe Deng, Huizhuo Yuan, Kaixuan Ji, and Quanquan Gu. Self-play fine-tuning converts weak language models to strong language models. *arXiv preprint arXiv:2401.01335*, 2024. 78, 87

[52] Zhanzhan Cheng, Fan Bai, Yunlu Xu, Gang Zheng, Shiliang Pu, and Shuigeng Zhou. Focusing attention: Towards accurate text recognition in natural images. In *ICCV*, 2017. 8

[53] Kyunghyun Cho, Aaron Courville, and Yoshua Bengio. Describing multimedia content using attention-based encoder-decoder networks. *TMM*, 17(11):1875–1886, 2015. 10

[54] Jan K Chorowski, Dzmitry Bahdanau, Dmitriy Serdyuk, Kyunghyun Cho, and Yoshua Bengio. Attention-based models for speech recognition. In *NIPS*, 2015. 10

[55] Kevin Clark, Minh-Thang Luong, Quoc V Le, and Christopher D Manning. Electra: Pre-training text encoders as discriminators rather than generators. In *ICLR*, 2020. 45

[56] Ekin D Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V Le. Randaugment: Practical automated data augmentation with a reduced search space. In *CVPR Workshop*, 2020. 45

[57] Zihang Dai, Zhilin Yang, Fan Yang, William W Cohen, and Ruslan Salakhutdinov. Good semi-supervised learning that requires a bad GAN. In *NeurIPS*, 2017. 21, 23

[58] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *CVPR*, 2005. 40

[59] Mostafa Dehghani, Josip Djolonga, Basil Mustafa, Piotr Padlewski, Jonathan Heek, Justin Gilmer, Andreas Peter Steiner, Mathilde Caron, Robert Geirhos, Ibrahim Alabdulmohsin, et al. Scaling vision transformers to 22 billion parameters. In *ICLR*, 2023. 102

[60] Marc Peter Deisenroth, Gerhard Neumann, Jan Peters, et al. A survey on policy search for robotics. *Foundations and Trends® in Robotics*, 2(1–2):1–142, 2013. 13

[61] Yuntian Deng, Anssi Kanervisto, Jeffrey Ling, and Alexander M Rush. Image-to-markup generation with coarse-to-fine attention. *ICML*, 2017. 10

[62] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *NAACL*, 2019. 38, 40, 41, 42, 43, 52

[63] Carl Doersch, Abhinav Gupta, and Alexei A Efros. Unsupervised visual representation

learning by context prediction. In *ICCV*, 2015. 24

[64] Patrick Doetsch, Michal Kozielski, and Hermann Ney. Fast and robust training of recurrent neural networks for offline handwriting recognition. In *ICFHR*, 2014. 8, 17

[65] Pedro Domingos. Knowledge acquisition from examples via multiple models. In *ICML*, 1997. 23

[66] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *ICLR*, 2021. x, 38, 39, 40, 41, 42, 43, 46, 50

[67] Zi-Yi Dou, Yichong Xu, Zhe Gan, Jianfeng Wang, Shuohang Wang, Lijuan Wang, Chenguang Zhu, Pengchuan Zhang, Lu Yuan, Nanyun Peng, Zicheng Liu, and Michael Zeng. An empirical study of training end-to-end vision-and-language transformers. *CVPR*, 2022. 1, 39, 40, 42, 43, 44, 49

[68] Zi-Yi Dou, Yichong Xu, Zhe Gan, Jianfeng Wang, Shuohang Wang, Lijuan Wang, Chenguang Zhu, Pengchuan Zhang, Lu Yuan, Nanyun Peng, et al. An empirical study of training end-to-end vision-and-language transformers. In *CVPR*, 2022. 47

[69] Nikita Dvornik, Cordelia Schmid, and Julien Mairal. Diversity with cooperation: Ensemble methods for few-shot classification. In *ICCV*, 2019. 23

[70] Salvador Espana-Boquera, Maria Jose Castro-Bleda, Jorge Gorbe-Moya, and Francisco Zamora-Martinez. Improving offline handwritten text recognition with hybrid hmm/ann models. *TPAMI*, 33(4):767–779, 2011. 15, 17

[71] Bernard Ghanem Fabian Caba Heilbron, Victor Escorcia and Juan Carlos Niebles. Activitynet: A large-scale video benchmark for human activity understanding. In *CVPR*, 2015. 86, 94

[72] Li Fei-Fei, Rob Fergus, and Pietro Perona. One-shot learning of object categories. *TPAMI*, 28(4):594–611, 2006. 20

[73] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *ICML*, 2017. 20, 23

[74] Hang Gao, Zheng Shou, Alireza, Zareian, Hanwang Zhang, and Shih-Fu Chang. Low-shot learning via covariance-preserving adversarial augmentation networks. In *NeurIPS*, 2018. 20, 23, 25

[75] François Garderes, Maryam Ziaeefard, Baptiste Abeloos, and Freddy Lecue. Conceptbert: Concept-aware representation for visual question answering. In *EMNLP*, 2020. 60, 62, 68

[76] Spyros Gidaris and Nikos Komodakis. Dynamic few-shot visual learning without forgetting. In *CVPR*, 2018. 23, 29, 30

[77] Spyros Gidaris and Nikos Komodakis. Generating classification weights with GNN denoising autoencoders for few-shot learning. In *CVPR*, 2019. 23, 29

[78] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations. In *ICLR*, 2018. 24

[79] Raghav Goyal, Samira Ebrahimi Kahou, Vincent Michalski, Joanna Materzynska, Susanne Westphal, Heuna Kim, Valentin Haenel, Ingo Fruend, Peter Yianilos, Moritz Mueller-Freitag, et al. The" something something" video database for learning and evaluating visual common sense. In *ICCV*, 2017. 86, 94

[80] Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In *CVPR*, 2017. 1, 44, 45, 52

[81] Alex Graves and Jürgen Schmidhuber. Offline handwriting recognition with multidimensional recurrent neural networks. In *NIPS*, 2009. 8

[82] Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *ICML*, 2006. 8, 14, 15

[83] Alex Graves, Abdel-rahman Mohamed, and Geoffrey Hinton. Speech recognition with deep recurrent neural networks. In *ICASSP*, 2013. 8

[84] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, Bilal Piot, Koray Kavukcuoglu, Rémi Munos, and Michal Valko. Bootstrap your own latent: A new approach to self-supervised learning. In *NeurIPS*, 2020. 24

[85] Emmanuele Grosicki and Haikal El-Abed. Icdar 2011-french handwriting recognition competition. In *ICDAR*, 2011. 17

[86] Emmanuèle Grosicki, Matthieu Carre, Jean-Marie Brodin, and Edouard Geoffrois. Rimes evaluation campaign for handwritten mail processing. In *ICFHR*, 2008. 15

[87] Anisha Gunjal, Jihan Yin, and Erhan Bas. Detecting and preventing hallucinations in large vision language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 18135–18143, 2024. 79

[88] Bharath Hariharan and Ross B. Girshick. Low-shot visual object recognition by shrinking and hallucinating features. In *ICCV*, 2017. 22, 23, 29

[89] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 52, 62

[90] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *CVPR*, 2020. 22, 24

[91] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. *CVPR*, 2022. x, 37, 39, 40, 41, 43, 45, 50, 52

[92] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015. 21, 23, 26

[93] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997. 8

[94] Arian Hosseini, Xingdi Yuan, Nikolay Malkin, Aaron Courville, Alessandro Sordoni, and Rishabh Agarwal. V-star: Training verifiers for self-taught reasoners. *arXiv preprint arXiv:2402.06457*, 2024. 78, 88

[95] Eduard H Hovy, Jaime G Carbonell, Hans Chalupsky, Anatole Gershman, Alex Hauptmann, Florian Metze, Teruko Mitamura, Zaid Sheikh, Ankit Dangi, Aditi Chaudhary, et al. Opera: Operations-oriented probabilistic extraction, reasoning, and analysis. In *TAC*, 2019. 1

[96] Po-Yao Huang, Ye Yuan, Zhenzhong Lan, Lu Jiang, and Alexander G Hauptmann. Video representation learning and latent concept mining for large-scale multi-label video classification. *arXiv preprint arXiv:1707.01408*, 2017. 1

[97] Zhicheng Huang, Zhaoyang Zeng, Bei Liu, Dongmei Fu, and Jianlong Fu. Pixel-bert: Aligning image pixels with text by deep multi-modal transformers. *arXiv preprint arXiv:2004.00849*, 2020. 39, 41, 44, 47, 48

[98] Zhicheng Huang, Zhaoyang Zeng, Yupan Huang, Bei Liu, Dongmei Fu, and Jianlong Fu. Seeing out of the box: End-to-end pre-training for vision-language representation learning. In *CVPR*, 2021. 39

[99] Drew A Hudson and Christopher D Manning. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In *CVPR*, 2019. 44

[100] Hamish Ivison, Yizhong Wang, Valentina Pyatkin, Nathan Lambert, Matthew Peters, Pradeep Dasigi, Joel Jang, David Wadden, Noah A Smith, Iz Beltagy, et al. Camels in a changing climate: Enhancing lm adaptation with tulu 2. *arXiv preprint arXiv:2311.10702*, 2023. 85

[101] Gautier Izacard and Edouard Grave. Leveraging passage retrieval with generative models for open domain question answering. In *EACL*, 2020. 62

[102] Yunseok Jang, Yale Song, Youngjae Yu, Youngjin Kim, and Gunhee Kim. Tgif-qa: Toward spatio-temporal reasoning in visual question answering. In *CVPR*, 2017. 79, 85, 86, 94

[103] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc V Le, Yunhsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *ICML*, 2021. 39, 61, 64

[104] Zequn Jie, Xiaodan Liang, Jiashi Feng, Xiaojie Jin, Wen Lu, and Shuicheng Yan. Tree-structured reinforcement learning for sequential object localization. In *NeurIPS*, 2016. 10

[105] Peng Jin, Ryuichi Takanobu, Caiwan Zhang, Xiaochun Cao, and Li Yuan. Chat-univi: Unified visual representation empowers large language models with image and video understanding. *arXiv preprint arXiv:2311.08046*, 2023. 84, 85, 87, 89

[106] Yang Jin, Zhicheng Sun, Kun Xu, Liwei Chen, Hao Jiang, Quzhe Huang, Chengru Song, Yuliang Liu, Di Zhang, Yang Song, et al. Video-lavit: Unified video-language pre-training with decoupled visual-motional tokenization. *arXiv preprint arXiv:2402.03161*, 2024. 79

[107] Jeff Johnson, Matthijs Douze, and Hervé Jégou. Billion-scale similarity search with gpus. *IEEE Transactions on Big Data*, 2019. 64

[108] Han Junwei, Yang Le, Zhang Dingwen, Chang Xiaojun, and Liang Xiaodan. Reinforcement cutting-agent learning for video object segmentation. In *CVPR*, 2018. 10

[109] Amita Kamath, Christopher Clark, Tanmay Gupta, Eric Kolve, Derek Hoiem, and Aniruddha Kembhavi. Webly supervised concept expansion for general purpose vision models. In *ECCV*, 2022. 52, 53

[110] Andrej Karpathy and Li Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In *CVPR*, 2015. 46

[111] Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. Dense passage retrieval for open-domain question answering. In *EMNLP*, 2020. 62, 66

[112] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. In *NeurIPS*, 2020. 22, 26

[113] Donghyun Kim, Kuniaki Saito, Tae-Hyun Oh, Bryan A Plummer, Stan Sclaroff, and Kate Saenko. CDSP: Cross-domain self-supervised pre-training. In *ICCV*, 2021. 23

[114] Jongmin Kim, Taesup Kim, Sungwoong Kim, and Chang D. Yoo. Edge-labeling graph neural network for few-shot learning. In *CVPR*, 2019. 23

[115] Wonjae Kim, Bokyung Son, and Ildoo Kim. Vilt: Vision-and-language transformer without convolution or region supervision. In *ICML*, 2021. 1, 37, 38, 39, 40, 42, 43, 44, 45, 46, 47, 49

[116] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. 2015. 16

[117] Gregory Koch, Richard Zemel, and Ruslan Salakhutdinov. Siamese neural networks for one-shot image recognition. In *ICML*, 2015. 23

[118] Michał Kozielski, Patrick Doetsch, and Hermann Ney. Improvements in rwth's system for off-line handwriting recognition. In *ICDAR*, 2013. 17

[119] Ivan Krasin, Tom Duerig, Neil Alldrin, Andreas Veit, Sami Abu-El-Haija, Serge Belongie, David Cai, Zheyun Feng, Vittorio Ferrari, Victor Gomes, Abhinav Gupta, Dhyanesh Narayanan, Chen Sun, Gal Chechik, and Kevin Murphy. Openimages: A public dataset for large-scale multi-label and multi-class image classification. *Dataset available from https://github.com/openimages*, 2016. 44, 46

[120] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *IJCV*, 123 (1):32–73, 2017. 44, 46

[121] Praveen Krishnan, Kartik Dutta, and CV Jawahar. Deep feature embedding for accurate recognition and retrieval of handwritten text. In *ICFHR*, 2016. 8

[122] John Langford and Tong Zhang. The epoch-greedy algorithm for multi-armed bandits with side information. In *NeurIPS*, 2008. 13

[123] Chen-Yu Lee and Simon Osindero. Recursive recurrent nets with attention modeling for ocr in the wild. In *CVPR*, 2016. 10

[124] Harrison Lee, Samrat Phatale, Hassan Mansoor, Kellie Lu, Thomas Mesnard, Colton Bishop, Victor Carbune, and Abhinav Rastogi. Rlaif: Scaling reinforcement learning from human feedback with ai feedback. *arXiv preprint arXiv:2309.00267*, 2023. 78

[125] Kenton Lee, Ming-Wei Chang, and Kristina Toutanova. Latent retrieval for weakly supervised open domain question answering. *arXiv preprint arXiv:1906.00300*, 2019. 67

[126] Kwonjoon Lee, Subhransu Maji, Avinash Ravichandran, and Stefano Soatto. Meta-learning with differentiable convex optimization. In *CVPR*, 2019. 23, 29, 32, 33

[127] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *ACL*, 2020. 63

[128] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. In *NeurIPS*, 2020. 62

[129] Gen Li, Nan Duan, Yuejian Fang, Daxin Jiang, and Ming Zhou. Unicoder-vl: A universal encoder for vision and language by cross-modal pre-training. *Proceedings of AAAI*, 2020. 61

[130] Guohao Li, Xin Wang, and Wenwu Zhu. Boosting visual question answering with context-aware knowledge aggregation. In *ACM MM*, 2020. 60, 62, 68

[131] Hongyang Li, David Eigen, Samuel Dodge, Matthew Zeiler, and Xiaogang Wang. Finding task-relevant features for few-shot learning by category traversal. In *CVPR*, 2019. 29

[132] Hui Li, Peng Wang, and Chunhua Shen. Towards end-to-end text spotting with convolutional recurrent neural networks. In *ICCV*, 2017. 8

[133] Junnan Li, Ramprasaath Selvaraju, Akhilesh Gotmare, Shafiq Joty, Caiming Xiong, and Steven Chu Hong Hoi. Align before fuse: Vision and language representation learning with momentum distillation. *NeurIPS*, 2021. 1, 39, 40, 43, 44, 45, 47, 48, 49

[134] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. *arXiv preprint arXiv:2201.12086*, 2022. 1, 40, 43, 47

[135] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. *arXiv preprint arXiv:2301.12597*, 2023. 79

[136] KunChang Li, Yinan He, Yi Wang, Yizhuo Li, Wenhai Wang, Ping Luo, Yali Wang, Limin Wang, and Yu Qiao. Videochat: Chat-centric video understanding. *arXiv preprint arXiv:2305.06355*, 2023. 79, 80, 84, 85

[137] Kunchang Li, Yali Wang, Yinan He, Yizhuo Li, Yi Wang, Yi Liu, Zun Wang, Jilan Xu, Guo Chen, Ping Luo, et al. Mvbench: A comprehensive multi-modal video understanding benchmark. *arXiv preprint arXiv:2311.17005*, 2023. 84

[138] Lei Li, Zhihui Xie, Mukai Li, Shunian Chen, Peiyi Wang, Liang Chen, Yazheng Yang, Benyou Wang, and Lingpeng Kong. Silkie: Preference distillation for large visual language models. *arXiv preprint arXiv:2312.10665*, 2023. 78, 79, 82

[139] Liunian Harold Li, Mark Yatskar, Da Yin, Cho-Jui Hsieh, and Kai-Wei Chang. Visualbert: A simple and performant baseline for vision and language. *arXiv preprint arXiv:1908.03557*, 2019. 38, 39, 47, 61

[140] Wei Li, Can Gao, Guocheng Niu, Xinyan Xiao, Hao Liu, Jiachen Liu, Hua Wu, and

Haifeng Wang. Unimo: Towards unified-modal understanding and generation via cross-modal contrastive learning. *ACL*, 2021. 39

[141] Wenbin Li, Lei Wang, Jinglin Xu, Jing Huo, Yang Gao, and Jiebo Luo. Revisiting local descriptor based image-to-class measure for few-shot learning. In *CVPR*, 2019. 31

[142] Wenbin Li, Jinglin Xu, Jing Huo, Lei Wang, Yang Gao, and Jiebo Luo. Distribution consistency based covariance metric networks for few-shot learning. In *AAAI*, 2019. 23

[143] Xiujun Li, Xi Yin, Chunyuan Li, Pengchuan Zhang, Xiaowei Hu, Lei Zhang, Lijuan Wang, Houdong Hu, Li Dong, Furu Wei, et al. Oscar: Object-semantics aligned pre-training for vision-language tasks. In *ECCV*, 2020. 1, 39, 40, 41, 44, 47, 65

[144] Yanwei Li, Chengyao Wang, and Jiaya Jia. Llama-vid: An image is worth 2 tokens in large language models. *arXiv preprint arXiv:2311.17043*, 2023. 84, 85, 87, 89

[145] Zhiqi Li, Guo Chen, Shilong Liu, Shihao Wang, Vibashan VS, Yishen Ji, Shiyi Lan, Hao Zhang, Yilin Zhao, Subhashree Radhakrishnan, et al. Eagle 2: Building post-training data strategies from scratch for frontier vision-language models. *arXiv preprint arXiv:2501.14818*, 2025. 102

[146] Zhizhong Li and Derek Hoiem. Learning without forgetting. *TPAMI*, 40(12):2935–2947, 2017. 23

[147] Xiaodan Liang, Lisa Lee, and Eric P Xing. Deep variation-structured reinforcement learning for visual relationship and attribute detection. In *CVPR*, 2017. 10

[148] Yann Lifchitz, Yannis Avrithis, Sylvaine Picard, and Andrei Bursuc. Dense classification and implanting for few-shot learning. In *CVPR*, 2019. 29

[149] Bin Lin, Yang Ye, Bin Zhu, Jiaxi Cui, Munan Ning, Peng Jin, and Li Yuan. Video-llava: Learning united visual representation by alignment before projection, 2023. 79, 85, 90

[150] Bin Lin, Bin Zhu, Yang Ye, Munan Ning, Peng Jin, and Li Yuan. Video-llava: Learning united visual representation by alignment before projection. *arXiv preprint arXiv:2311.10122*, 2023. 84, 85, 87, 89

[151] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014. 1, 44, 46

[152] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. *arXiv preprint arXiv:2310.03744*, 2023. 79, 102

[153] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *arXiv preprint arXiv:2304.08485*, 2023. 78, 79

[154] Hugo Liu and Push Singh. Conceptnet—a practical commonsense reasoning tool-kit. *BT technology journal*, 22(4):211–226, 2004. 60

[155] Ruyang Liu, Chen Li, Yixiao Ge, Ying Shan, Thomas H Li, and Ge Li. One for all: Video conversation is feasible without video instruction tuning. *arXiv preprint arXiv:2309.15785*, 2023. 79, 84

[156] Yanbin Liu, Juho Lee, Minseop Park, Saehoon Kim, Eunho Yang, Sung Ju Hwang, and Yi Yang. Learning to propagate labels: Transductive propagation network for few-shot learning. In *ICLR*, 2019. 29

[157] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *CVPR*, 2021. 50, 51

[158] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019. 45, 67

[159] Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. In *NeurIPS*, 2019. 37, 38, 39, 40, 41, 43, 46, 47, 52, 61

[160] Man Luo, Yankai Zeng, Pratyay Banerjee, and Chitta Baral. Weakly-supervised visual-retriever-reader for knowledge-based question answering. In *EMNLP*, 2021. 61, 62, 64, 68

[161] Ruipu Luo, Ziwang Zhao, Min Yang, Junwei Dong, Minghui Qiu, Pengcheng Lu, Tao Wang, and Zhongyu Wei. Valley: Video assistant with large language model enhanced ability. *arXiv preprint arXiv:2306.07207*, 2023. 79

[162] Minh-Thang Luong, Hieu Pham, and Christopher D Manning. Effective approaches to attention-based neural machine translation. *EMNLP*, 2015. 10, 14

[163] Muhammad Maaz, Hanoona Rasheed, Salman Khan, and Fahad Shahbaz Khan. Video-chatgpt: Towards detailed video understanding via large vision and language models. *arXiv preprint arXiv:2306.05424*, 2023. 79, 84, 85, 87, 89

[164] Sabri A Mahmoud, Irfan Ahmad, Mohammad Alshayeb, Wasfi G Al-Khatib, Mohammad Tanvir Parvez, Gernot A Fink, Volker Märgner, and Haikal El Abed. Khatt: Arabic offline handwritten text database. In *ICFHR*, 2012. 15

[165] Puneet Mangla, Nupur Kumari, Abhishek Sinha, Mayank Singh, Balaji Krishnamurthy, and Vineeth N Balasubramanian. Charting the right manifold: Manifold mixup for few-shot learning. In *WACV*, 2020. 32, 33

[166] Kenneth Marino, Mohammad Rastegari, Ali Farhadi, and Roozbeh Mottaghi. Ok-vqa: A

visual question answering benchmark requiring external knowledge. In *CVPR*, 2019. 1, 60, 61, 67, 68

[167] Kenneth Marino, Xinlei Chen, Devi Parikh, Abhinav Gupta, and Marcus Rohrbach. Krisp: Integrating implicit and symbolic knowledge for open-domain knowledge-based vqa. In *CVPR*, 2021. 52, 53, 60, 61, 62, 63, 64, 67, 68

[168] U-V Marti and Horst Bunke. The iam-database: an english sentence database for offline handwriting recognition. *IJDAR*, 5(1):39–46, 2002. 15

[169] Farès Menasri, Jérôme Louradour, Anne-Laure Bianne-Bernard, and Christopher Kermorvant. The a2ia french handwriting recognition system at the rimes-icdar2011 competition. In *Document Recognition and Retrieval XIX*, volume 8297, page 82970Y. International Society for Optics and Photonics, 2012. 17

[170] Ronaldo Messina and Christopher Kermorvant. Over-generative finite state transducer n-gram for out-of-vocabulary word recognition. In *DAS Workshop*, 2014. 17

[171] Seyed Iman Mirzadeh, Mehrdad Farajtabar, Ang Li, Nir Levine, Akihiro Matsukawa, and Hassan Ghasemzadeh. Improved knowledge distillation via teacher assistant. In *AAAI*, 2020. 23

[172] Takeru Miyato, Shin-ichi Maeda, Masanori Koyama, and Shin Ishii. Virtual adversarial training: A regularization method for supervised and semi-supervised learning. *TPAMI*, 41 (8):1979–1993, 2018. 26

[173] Volodymyr Mnih, Nicolas Heess, Alex Graves, et al. Recurrent models of visual attention. In *NeurIPS*, 2014. 10

[174] Ron Mokady, Amir Hertz, and Amit H Bermano. Clipcap: Clip prefix for image captioning. *arXiv preprint arXiv:2111.09734*, 2021. 52, 53

[175] Medhini Narasimhan, Svetlana Lazebnik, and Alexander Schwing. Out of the box: Reasoning with graph convolution nets for factual visual question answering. *NeurIPS*, 2018. 62

[176] Mehdi Noroozi and Paolo Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. In *ECCV*, 2016. 24

[177] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018. 24

[178] Vicente Ordonez, Girish Kulkarni, and Tamara Berg. Im2text: Describing images using 1 million captioned photographs. *NeurIPS*, 2011. 1, 44

[179] Boris Oreshkin, Pau Rodríguez López, and Alexandre Lacoste. TADAM: Task dependent

adaptive metric for improved few-shot learning. In *NeurIPS*, 2018. 23, 29

[180] Nobuyuki Otsu. A threshold selection method from gray-level histograms. *IEEE transactions on systems, man, and cybernetics*, 9(1):62–66, 1979. 16

[181] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744, 2022. 78

[182] Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. On the difficulty of training recurrent neural networks. In *ICML*, 2013. 16

[183] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. Context encoders: Feature learning by inpainting. In *CVPR*, 2016. 24

[184] Vu Pham, Théodore Bluche, Christopher Kermorvant, and Jérôme Louradour. Dropout improves recurrent neural networks for handwriting recognition. In *ICFHR*, 2014. 17

[185] Bryan A Plummer, Liwei Wang, Chris M Cervantes, Juan C Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In *ICCV*, 2015. 1, 44, 46

[186] Arik Poznanski and Lior Wolf. Cnn-n-gram for handwriting word recognition. In *CVPR*, 2016. 8, 9, 15, 17

[187] Di Qi, Lin Su, Jia Song, Edward Cui, Taroon Bharti, and Arun Sacheti. Imagebert: Cross-modal pre-training with large-scale weak-supervised image-text data. *arXiv preprint arXiv:2001.07966*, 2020. 39

[188] Siyuan Qiao, Chenxi Liu, Wei Shen, and Alan L Yuille. Few-shot image recognition by predicting parameters from activations. In *CVPR*, 2018. 29

[189] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, 2021. 39, 52, 61, 64, 71

[190] Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems*, 36, 2024. 78

[191] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *JMLR*, 21(140):1–67, 2020. URL `http://jmlr.org/papers/v21/20-074.html`. 63, 67

[192] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *ICML*, 2021. xvi, 40, 50, 52

[193] Sachin Ravi and Hugo Larochelle. Optimization as a model for few-shot learning. In *ICLR*, 2017. 22, 23, 28

[194] Avinash Ravichandran, Rahul Bhotika, and Stefano Soatto. Few-shot learning with embedded class models and shot-free meta training. In *ICCV*, 2019. 29

[195] Mengye Ren, Eleni Triantafillou, Sachin Ravi, Jake Snell, Kevin Swersky, Joshua B Tenenbaum, Hugo Larochelle, and Richard S Zemel. Meta-learning for semi-supervised few-shot classification. In *ICLR*, 2018. 22, 28

[196] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *NeurIPS*, 2015. 39, 46, 62

[197] Steven J Rennie, Etienne Marcheret, Youssef Mroueh, Jarret Ross, and Vaibhava Goel. Self-critical sequence training for image captioning. *CVPR*, 2017. 13

[198] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *IJCV*, 115(3):211–252, 2015. 37, 38

[199] Andrei A Rusu, Dushyant Rao, Jakub Sygnowski, Oriol Vinyals, Razvan Pascanu, Simon Osindero, and Raia Hadsell. Meta-learning with latent embedding optimization. In *ICLR*, 2019. 23, 29

[200] Kuniaki Saito, Donghyun Kim, Stan Sclaroff, and Kate Saenko. Universal domain adaptation through self supervision. In *NeurIPS*, 2020. 23

[201] Joan Andreu Sanchez, Veronica Romero, Alejandro H Toselli, and Enrique Vidal. Icfhr2016 competition on handwritten text recognition on the read dataset. In *ICFHR*, 2016. 8

[202] Victor Garcia Satorras and Joan Bruna Estrach. Few-shot learning with graph neural networks. In *ICLR*, 2018. 23

[203] Jürgen Schmidhuber. Evolutionary principles in self-referential learning. *On learning how to learn: The meta-meta-... hook. Diploma thesis, Institut f. Informatik, Tech. Univ. Munich*, 1987. 20

[204] Eli Schwartz, Leonid Karlinsky, Joseph Shtok, Sivan Harary, Mattias Marder, Abhishek Kumar, Rogerio Feris, Raja Giryes, and Alex M. Bronstein. Delta-encoder: An effective sample synthesis method for few-shot object recognition. In *NeurIPS*, 2018. 20, 23, 25

[205] Jonathan Schwarz, Jelena Luketina, Wojciech M Czarnecki, Agnieszka Grabska-Barwinska,

Yee Whye Teh, Razvan Pascanu, and Raia Hadsell. Progress & compress: A scalable framework for continual learning. In *ICML*, 2018. 23

[206] Dustin Schwenk, Apoorv Khandelwal, Christopher Clark, Kenneth Marino, and Roozbeh Mottaghi. A-okvqa: A benchmark for visual question answering using world knowledge. *arXiv preprint arXiv:2206.01718*, 2022. 1, 52

[207] Shuai Shao, Zeming Li, Tianyuan Zhang, Chao Peng, Gang Yu, Xiangyu Zhang, Jing Li, and Jian Sun. Objects365: A large-scale, high-quality dataset for object detection. In *ICCV*, 2019. 46

[208] Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *ACL*, 2018. 44

[209] Baoguang Shi, Xiang Bai, and Cong Yao. An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition. *TPAMI*, 39 (11):2298–2304, 2017. 8, 10, 12, 16, 17, 18

[210] Konstantin Shmelkov, Cordelia Schmid, and Karteek Alahari. How good is my GAN? In *ECCV*, 2018. 21, 23

[211] Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh, and Marcus Rohrbach. Towards vqa models that can read. In *CVPR*, 2019. 1

[212] Amanpreet Singh, Ronghang Hu, Vedanuj Goswami, Guillaume Couairon, Wojciech Galuba, Marcus Rohrbach, and Douwe Kiela. Flava: A foundational language and vision alignment model. In *CVPR*, 2022. 1

[213] Jake Snell, Kevin Swersky, and Richard S. Zemel. Prototypical networks for few-shot learning. In *NeurIPS*, 2017. 20, 23, 25, 29, 31

[214] Weijie Su, Xizhou Zhu, Yue Cao, Bin Li, Lewei Lu, Furu Wei, and Jifeng Dai. Vl-bert: Pre-training of generic visual-linguistic representations. In *ICLR*, 2020. 39, 40, 61

[215] Jorge Sueiras, Victoria Ruiz, Angel Sanchez, and Jose F Velez. Offline continuous handwriting recognition using sequence to sequence neural networks. *Neurocomputing*, 289: 119–128, 2018. 9

[216] Alane Suhr, Stephanie Zhou, Ally Zhang, Iris Zhang, Huajun Bai, and Yoav Artzi. A corpus for reasoning about natural language grounded in photographs. *ACL*, 2018. 44, 45

[217] Chen Sun, Abhinav Shrivastava, Saurabh Singh, and Abhinav Gupta. Revisiting unreasonable effectiveness of data in deep learning era. In *ICCV*, 2017. 32

[218] Qianru Sun, Yaoyao Liu, Tat-Seng Chua, and Bernt Schiele. Meta-transfer learning for

few-shot learning. In *CVPR*, 2019. 29

[219] Zhiqing Sun, Sheng Shen, Shengcao Cao, Haotian Liu, Chunyuan Li, Yikang Shen, Chuang Gan, Liang-Yan Gui, Yu-Xiong Wang, Yiming Yang, et al. Aligning large multimodal models with factually augmented rlhf. *arXiv preprint arXiv:2309.14525*, 2023. 78, 79, 82

[220] Zhiqing Sun, Yikang Shen, Hongxin Zhang, Qinhong Zhou, Zhenfang Chen, David Cox, Yiming Yang, and Chuang Gan. Salmon: Self-alignment with principle-following reward models. *arXiv preprint arXiv:2310.05910*, 2023. 78

[221] Flood Sung, Yongxin Yang, Li Zhang, Tao Xiang, Philip HS Torr, and Timothy M Hospedales. Learning to compare: Relation network for few-shot learning. In *CVPR*, 2018. 23, 29

[222] Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*, volume 1. MIT press Cambridge, 1998. 9, 13

[223] Hao Tan and Mohit Bansal. Lxmert: Learning cross-modality encoder representations from transformers. In *EMNLP*, 2019. 37, 38, 39, 40, 45, 47, 52, 61

[224] Yee Teh, Victor Bapst, Wojciech M Czarnecki, John Quan, James Kirkpatrick, Raia Hadsell, Nicolas Heess, and Razvan Pascanu. Distral: Robust multitask reinforcement learning. In *NeurIPS*, 2017. 23

[225] Sebastian Thrun and Lorien Pratt. *Learning to learn*. Springer Science & Business Media, 2012. 20, 23

[226] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. In *ECCV*, 2020. 24

[227] Yonglong Tian, Yue Wang, Dilip Krishnan, Joshua B Tenenbaum, and Phillip Isola. Rethinking few-shot image classification: A good embedding is all you need? In *ECCV*, 2020. 29

[228] Shengbang Tong, Zhuang Liu, Yuexiang Zhai, Yi Ma, Yann LeCun, and Saining Xie. Eyes wide shut? exploring the visual shortcomings of multimodal llms. In *CVPR*, 2024. 102

[229] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *ICML*, 2021. 50

[230] Laurens van der Maaten and Geoffrey Hinton. Visualizing high-dimensional data using t-SNE. *JMLR*, 9(11):2579–2605, 2008. 34

[231] Pavan Kumar Anasosalu Vasu, Fartash Faghri, Chun-Liang Li, Cem Koc, Nate True, Albert Antony, Gokul Santhanam, James Gabriel, Peter Grasch, Oncel Tuzel, et al. Fastvlm:

Efficient vision encoding for vision language models. *arXiv preprint arXiv:2412.13303*, 2024. 103

[232] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017. 66

[233] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Koray Kavukcuoglu, and Daan Wierstra. Matching networks for one shot learning. In *NeurIPS*, 2016. 20, 22, 23, 24, 28, 29, 31

[234] Denny Vrandecic and Markus Krotzsch. Wikidata: a free collaborative knowledgebase. *Communications of the ACM*, 2014. 64, 65

[235] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. The Caltech-UCSD Birds-200-2011 Dataset. Technical Report CNS-TR-2011-001, California Institute of Technology, 2011. 22, 29

[236] Jianfeng Wang and Xiaolin Hu. Gated recurrent convolution neural network for ocr. In *NIPS*, 2017. 8

[237] Peng Wang, Qi Wu, Chunhua Shen, Anthony Dick, and Anton Van Den Hengel. Fvqa: Fact-based visual question answering. *TPAMI*, 40(10):2413–2427, 2017. 61, 62

[238] Peng Wang, Qi Wu, Chunhua Shen, Anton van den Hengel, and Anthony Dick. Explicit knowledge-based reasoning for visual question answering. In *IJCAI*, 2017. 60, 62

[239] Peng Wang, An Yang, Rui Men, Junyang Lin, Shuai Bai, Zhikang Li, Jianxin Ma, Chang Zhou, Jingren Zhou, and Hongxia Yang. Unifying architectures, tasks, and modalities through a simple sequence-to-sequence learning framework. In *ICML*, 2022. 47

[240] Xinlong Wang, Xiaosong Zhang, Zhengxiong Luo, Quan Sun, Yufeng Cui, Jinsheng Wang, Fan Zhang, Yueze Wang, Zhen Li, Qiying Yu, et al. Emu3: Next-token prediction is all you need. *arXiv preprint arXiv:2409.18869*, 2024. 102

[241] Yi Wang, Yinan He, Yizhuo Li, Kunchang Li, Jiashuo Yu, Xin Ma, Xinhao Li, Guo Chen, Xinyuan Chen, Yaohui Wang, et al. Internvid: A large-scale video-text dataset for multimodal understanding and generation. *arXiv preprint arXiv:2307.06942*, 2023. 79

[242] Yu-Xiong Wang and Martial Hebert. Learning to learn: Model regression networks for easy small sample learning. In *ECCV*, 2016. 20, 21

[243] Yu-Xiong Wang, Deva Ramanan, and Martial Hebert. Learning to model the tail. In *NeurIPS*, 2017. 21, 32

[244] Yu-Xiong Wang, Ross B. Girshick, Martial Hebert, and Bharath Hariharan. Low-shot learning from imaginary data. In *CVPR*, 2018. ix, xv, 20, 22, 23, 25, 29, 30, 31, 32, 33, 34

[245] Zirui Wang, Jiahui Yu, Adams Wei Yu, Zihang Dai, Yulia Tsvetkov, and Yuan Cao.

Simvlm: Simple visual language model pretraining with weak supervision. *arXiv preprint arXiv:2108.10904*, 2021. 2

[246] Chen Wei, Haoqi Fan, Saining Xie, Chao-Yuan Wu, Alan Yuille, and Christoph Feichtenhofer. Masked feature prediction for self-supervised visual pre-training. *arXiv preprint arXiv:2112.09133*, 2021. 40, 50

[247] Yi Wei, Xinyu Pan, Hongwei Qin, and Junjie Yan. Quantization mimic: Towards very tiny CNN for object detection. In *ECCV*, 2018. 23

[248] Peter West, Chandra Bhagavatula, Jack Hessel, Jena D. Hwang, Liwei Jiang, Ronan Le Bras, Ximing Lu, Sean Welleck, and Yejin Choi. Symbolic knowledge distillation: from general language models to commonsense models. In *ArXiv*, 2021. 62

[249] Curtis Wigington, Seth Stewart, Brian Davis, Bill Barrett, Brian Price, and Scott Cohen. Data augmentation for recognition of handwritten words and lines using a cnn-lstm network. In *ICDAR*, 2017. 16, 17

[250] Jialin Wu, Jiasen Lu, Ashish Sabharwal, and Roozbeh Mottaghi. Multi-modal answer validation for knowledge-based vqa. In *AAAI*, 2022. 60, 61, 62, 63, 68

[251] Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*, 2016. 41

[252] Zhirong Wu, Yuanjun Xiong, Stella X Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *CVPR*, 2018. 22, 24

[253] Zuxuan Wu, Tushar Nagarajan, Abhishek Kumar, Steven Rennie, Larry S Davis, Kristen Grauman, and Rogerio Feris. Blockdrop: Dynamic inference paths in residual networks. 2018. 10, 13

[254] Yongqin Xian, Tobias Lorenz, Bernt Schiele, and Zeynep Akata. Feature generating networks for zero-shot learning. In *CVPR*, 2018. 23

[255] Yongqin Xian, Saurabh Sharma, Bernt Schiele, and Zeynep Akata. f-VAEGAN-D2: A feature generating framework for any-shot learning. In *CVPR*, 2019. 20, 23

[256] Junbin Xiao, Xindi Shang, Angela Yao, and Tat-Seng Chua. Next-qa: Next phase of question-answering to explaining temporal actions. In *CVPR*, 2021. 85

[257] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *CVPR*, 2017. 46

[258] Zhenda Xie, Zheng Zhang, Yue Cao, Yutong Lin, Jianmin Bao, Zhuliang Yao, Qi Dai, and

Han Hu. Simmim: A simple framework for masked image modeling. In *CVPR*, 2022. 40, 50

[259] Jiaolong Xu, Peng Wang, Heng Yang, and Antonio M. López. Training a binary weight object detector by knowledge transfer for autonomous driving. In *ICRA*, 2019. 23

[260] Jun Xu, Tao Mei, Ting Yao, and Yong Rui. Msr-vtt: A large video description dataset for bridging video and language. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5288–5296, 2016. 79, 85, 86, 94

[261] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *ICML*, 2015. 10

[262] Zheng Xu, Yen-Chang Hsu, and Jiawei Huang. Training shallow and thin networks for acceleration via knowledge distillation with conditional adversarial networks. In *ICLR*, 2018. 23

[263] Hongwei Xue, Yupan Huang, Bei Liu, Houwen Peng, Jianlong Fu, Houqiang Li, and Jiebo Luo. Probing inter-modality: Visual parsing with self-attention for vision-and-language pre-training. *NeurIPS*, 2021. 39, 43, 47, 48

[264] Antoine Yang, Antoine Miech, Josef Sivic, Ivan Laptev, and Cordelia Schmid. Zero-shot video question answering via frozen bidirectional language models. *NeurIPS*, 2022. 84, 85

[265] Zhengyuan Yang, Zhe Gan, Jianfeng Wang, Xiaowei Hu, Yumao Lu, Zicheng Liu, and Lijuan Wang. An empirical study of gpt-3 for few-shot knowledge-based vqa. In *AAAI*, 2022. 62, 65, 68

[266] Han-Jia Ye, Hexiang Hu, De-Chuan Zhan, and Fei Sha. Few-shot learning via embedding adaptation with set-to-set functions. In *CVPR*, 2020. 29

[267] Mang Ye, Xu Zhang, Pong C Yuen, and Shih-Fu Chang. Unsupervised embedding learning via invariant and spreading instance feature. In *CVPR*, 2019. 24

[268] Peter Young, Alice Lai, Micah Hodosh, and Julia Hockenmaier. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *TACL*, 2:67–78, 2014. 44

[269] Jiahui Yu, Zirui Wang, Vijay Vasudevan, Legg Yeung, Mojtaba Seyedhosseini, and Yonghui Wu. Coca: Contrastive captioners are image-text foundation models. *arXiv preprint arXiv:2205.01917*, 2022. 2, 47

[270] Zhou Yu, Dejing Xu, Jun Yu, Ting Yu, Zhou Zhao, Yueting Zhuang, and Dacheng Tao. Activitynet-qa: A dataset for understanding complex web videos via question answering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 9127–

9134, 2019. 79

[271] Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stéphane Deny. Barlow twins: Self-supervised learning via redundancy reduction. 2021. 24

[272] Yan Zeng, Xinsong Zhang, and Hang Li. Multi-grained vision language pre-training: Aligning texts with visual concepts. In *ICML*, 2022. 47, 48

[273] Chi Zhang, Yujun Cai, Guosheng Lin, and Chunhua Shen. DeepEMD: Few-shot image classification with differentiable earth mover's distance and structured classifiers. In *CVPR*, 2020. 23, 29, 30, 31, 32, 33

[274] Hang Zhang, Xin Li, and Lidong Bing. Video-llama: An instruction-tuned audio-visual language model for video understanding. *arXiv preprint arXiv:2306.02858*, 2023. 79, 84, 85

[275] Hongguang Zhang, Jing Zhang, and Piotr Koniusz. Few-shot learning via saliency-guided hallucination of samples. In *CVPR*, 2019. 20

[276] Pengchuan Zhang, Xiujun Li, Xiaowei Hu, Jianwei Yang, Lei Zhang, Lijuan Wang, Yejin Choi, and Jianfeng Gao. Vinvl: Revisiting visual representations in vision-language models. In *CVPR*, 2021. 1, 39, 40, 41, 44, 46, 47

[277] Renrui Zhang, Jiaming Han, Aojun Zhou, Xiangfei Hu, Shilin Yan, Pan Lu, Hongsheng Li, Peng Gao, and Yu Qiao. Llama-adapter: Efficient fine-tuning of language models with zero-init attention. *arXiv preprint arXiv:2303.16199*, 2023. 84

[278] Richard Zhang, Phillip Isola, and Alexei A Efros. Colorful image colorization. In *ECCV*, 2016. 24

[279] Richard Zhang, Phillip Isola, and Alexei A Efros. Split-brain autoencoders: Unsupervised learning by cross-channel prediction. In *CVPR*, 2017. 24

[280] Ruixiang Zhang, Tong Che, Zoubin Ghahramani, Yoshua Bengio, and Yangqiu Song. MetaGAN: An adversarial approach to few-shot learning. In *NeurIPS*, 2018. 20, 23

[281] Weilin Zhang and Yu-Xiong Wang. Hallucination improves few-shot object detection. In *CVPR*, 2021. 20

[282] Yuanhan Zhang, Bo Li, haotian Liu, Yong jae Lee, Liangke Gui, Di Fu, Jiashi Feng, Ziwei Liu, and Chunyuan Li. Llava-next: A strong zero-shot video understanding model, April 2024. URL `https://llava-vl.github.io/blog/2024-04-30-llava-next-video/`. 102

[283] Zhao Zhong, Junjie Yan, Wei Wu, Jing Shao, and Liu Cheng-Lin. Practical block-wise neural network architecture generation. In *CVPR*, 2018. 10

[284] Chunting Zhou, Lili Yu, Arun Babu, Kushal Tirumala, Michihiro Yasunaga, Leonid Shamis, Jacob Kahn, Xuezhe Ma, Luke Zettlemoyer, and Omer Levy. Transfusion: Predict the next token and diffuse images with one multi-modal model. *arXiv preprint arXiv:2408.11039*, 2024. 102

[285] Xuhui Zhou, Hao Zhu, Leena Mathur, Ruohong Zhang, Haofei Yu, Zhengyang Qi, Louis-Philippe Morency, Yonatan Bisk, Daniel Fried, Graham Neubig, et al. Sotopia: Interactive evaluation for social intelligence in language agents. *arXiv preprint arXiv:2310.11667*, 2023. 83

[286] Bin Zhu, Bin Lin, Munan Ning, Yang Yan, Jiaxi Cui, HongFa Wang, Yatian Pang, Wenhao Jiang, Junwu Zhang, Zongwei Li, et al. Languagebind: Extending video-language pretraining to n-modality by language-based semantic alignment. *arXiv preprint arXiv:2310.01852*, 2023. 86, 94