Advancing Toward Temporal and Commonsense Reasoning in Vision-Language Learning

by

İlker Kesen

A Dissertation Submitted to the
Graduate School of Sciences and Engineering
in Partial Fulfillment of the Requirements for
the Degree of

Doctor of Philosophy

in

Computer Science and Engineering

KOÇ ÜNİVERSİTESİ

September 15, 2023

Advancing Toward Temporal and Commonsense Reasoning in Vision-Language Learning

Koç University

Graduate School of Sciences and Engineering

This is to certify that I have examined this copy of a doctoral dissertation by

İlker Kesen

and have found that it is complete and satisfactory in all respects,
and that any and all revisions required by the final
examining committee have been made.

Committee Members:

_____

Prof. Deniz Yuret (Advisor)

_____

Assoc. Prof. Aykut Erdem

_____

Assoc. Prof. Tilbe Göksun

_____

Assoc. Prof. Erkut Erdem

_____

Assoc. Prof. Arzucan Özgür

Date: _____

ABSTRACT

Advancing Toward Temporal and Commonsense Reasoning in Vision-Language Learning
İlker Kesen
Doctor of Philosophy in Computer Science and Engineering
September 15, 2023

Humans learn to ground language to the world through experience, primarily visual observations. Devising natural language processing (NLP) approaches that can reason in a similar sense to humans is a long-standing objective of the artificial intelligence community. Recently, transformer models exhibited remarkable performance on numerous NLP tasks. This is followed by breakthroughs in vision-language (V&L) tasks, like image captioning and visual question answering, which require connecting language to the visual world. These successes of transformer models encouraged the V&L community to pursue more challenging directions, most notably temporal and commonsense reasoning.

This thesis focuses on V&L problems that require either temporal reasoning, commonsense reasoning, or both simultaneously. Temporal reasoning is the ability to reason over time. In the context of V&L, this means going beyond static images, i.e., processing videos. Commonsense reasoning requires capturing the implicit general knowledge about the world surrounding us and making an accurate judgment using this knowledge within a particular context. This thesis comprises four distinct studies that connect language and vision by exploring various aspects of temporal and commonsense reasoning.

Before advancing to these challenging directions, (i) we first focus on the localization stage: We experiment with a model that enables systematic evaluation of how language-conditioning should affect the bottom-up and the top-down visual processing branches. We show that conditioning the bottom-up branch on language is crucial to ground visual concepts like colors and object categories. (ii) Next, we investigate whether the existing video-language models thrive in answering questions about complex dynamic scenes. We choose the CRAFT benchmark as our test

bed and show that the state-of-the-art video-language models fall behind human performance by a large margin, failing to process dynamic scenes proficiently. (iii) In the third study, we develop a zero-shot video-language evaluation benchmark to evaluate the language understanding abilities of pretrained video-language models. Our experiments reveal that the current video-language models are no better than the vision-language models, processing static images as input in processing daily dynamic actions. (iv) In the last study, we work on a figurative language understanding problem called euphemism detection. Euphemisms tone down expressions about sensitive or unpleasant issues. The ambiguous nature of euphemistic terms makes it challenging to detect their actual meaning within a context where commonsense knowledge and reasoning are necessities. We show that incorporating additional textual and visual knowledge in low-resource settings is beneficial to detect euphemistic terms. Nonetheless, our findings on these four studies still demonstrate a substantial gap between current V&L models' abilities and human cognition.

ÖZETÇE

Görü-Dil Öğreniminde Zamansal ve
Sağduyulu Muhakemeye Doğru İlerleme
İlker Kesen
Bilgisayar Bilimleri ve Mühendisliği, Doktora
15 Eylül 2023

İnsanlar, başta gözlemler olmak üzere deneyimler yoluyla dili dünyaya dayandırmayı öğrenirler. İnsanlara benzer şekilde akıl yürütebilen doğal dil işleme (NLP) yaklaşımları geliştirmek, yapay zeka topluluğunun uzun süredir devam eden bir hedefidir. Son zamanlarda, dönüştürücü modeller çok sayıda NLP görevinde kayda değer performans ortaya koymuştur. Bunu, görüntü altyazılama ve görsel soru yanıtlama gibi, dili görsel dünyaya bağlamayı gerektiren görü-dil (V&L) görevlerindeki atılımlar izledi. Dönüştürücü modellerin bu başarıları, V&L topluluğunu, özellikle zamansal ve sağduyulu akıl yürütme gibi daha zorlu yönleri takip etmeye yönlendirmiştir.

Bu tez, zamansal muhakeme, sağduyulu muhakeme ya da her ikisini aynı anda gerektiren V&L problemlerine odaklanmaktadır. Zamansal akıl yürütme, zaman içinde akıl yürütme yeteneğidir. V&L bağlamında bu, durağan görüntülerin ötesine geçmek, yani videoları işlemek anlamına gelmektedir. Sağduyulu muhakeme, bizi çevreleyen dünya hakkındaki örtük genel bilgiyi yakalamayı ve bu bilgiyi belirli bir içerik dahilinde kullanarak doğru bir yargıya varmayı gerektirir. Bu tez, zamansal ve sağduyulu muhakemenin çeşitli yönlerini araştırarak dil ve görüyü birbirine bağlayan dört farklı çalışmadan oluşmaktadır.

Bu zorlu yönlere geçmeden önce, (i) ilk olarak konumlandırma aşamasına odaklanılmaktadır: Dil koşullandırmasının aşağıdan yukarıya ve yukarıdan aşağıya görsel işleme dallarını nasıl etkilemesi gerektiğinin sistematik olarak değerlendirilmesini sağlayan bir modelle çalışılmıştır. Aşağıdan yukarıya olan dalın dile koşullanmasının, renkler ve nesne kategorileri gibi görsel kavramları temellendirmek için çok önemli olduğunu gösterilmiştir. (ii) Sonrasında, mevcut video-dil modellerinin karmaşık dinamik sahnelerle ilgili soruları yanıtlamada başarılı olup olmadığı araştırılmıştır. Test ortamı olarak CRAFT veri kümesi tercih edilmiş ve son teknoloji video-dil

modellerinin dinamik sahneleri yetkin bir şekilde işleyemeyerek büyük bir farkla insan performansının gerisinde kaldığı gösterilmiştir. (iii) Üçüncü çalışmada, önceden eğitilmiş video-dil modellerinin dil anlama yeteneklerini değerlendirmek için sıfır atış video-dil değerlendirme ölçütü geliştiriyoruz. Yapılan deneyler, mevcut video-dil modellerinin, günlük dinamik eylemlerin işlenmesinde girdi olarak statik görüntüleri işleyen görme-dil modellerinden daha iyi olmadığını ortaya koymaktadır. (iv) Son çalışmada, örtmece algılama adı verilen mecazi bir dil anlama problemi üzerinde çalışılmıştır. Örtmeceler, hassas veya hoş olmayan konularla ilgili ifadeleri yumuşatır. Örtmece terimlerin müphem doğası, sağduyu bilgisinin ve sağduyulu muhakemenin gerekli olduğu bir durumda gerçek anlamlarının tespit edilmesini zorlaştırmaktadır. Düşük kaynaklı ortamlarda ek metinsel ve görsel bilginin dahil edilmesinin örtmece terimlerin tespit edilmesinde faydalı olduğunu gösterilmiştir. Bununla birlikte, bu dört çalışma ile ilgili elde edilen bulgular, mevcut V&L modellerinin yetenekleri ile insan muhakemesi arasında hala ciddi bir uçurum olduğunu göstermektedir.

ACKNOWLEDGMENTS

First and foremost, I would like to thank my advisor, Deniz Yuret, for his guidance and support throughout my PhD journey. He consistently challenged me to think outside the box and pushed the boundaries of my conceptual thinking skills, eventually making me a more proficient researcher. I learned from him a lot, especially about how to debug ideas and how to convey them in a clear and concise way.

I present my uttermost gratitude to my thesis jury for their valuable feedback. I want to thank each jury member individually. I want to start with Tilbe Göksun, who has continuously brought a positive and encouraging presence. Her expertise in cognitive science and wisdom allowed me to see the big picture in AI. I want to thank Arzucan Özgür for her valuable feedback on the thesis and advice on the future during this period.

I do not know how to express my gratefulness for Aykut Erdem and Erkut Erdem. This thesis would not exist without them. They are such brilliant scientists, supervisors, and persons. I admire their ability to consistently stay ahead of the curve in this rapidly evolving research field. They had always been encouraging during my studies, which allowed me to engage more with the community. I am very fortunate to have worked with them.

I also want to thank the people I collaborated with, especially Iacer Calixto. Iacer is also an excellent researcher, like each member of my jury. He hosted me for two months in his institute, Amsterdam Medical Center, which was an invaluable experience. During my visit, I had the opportunity to interact with researchers from my field and other fields. Additionally, I would like to thank Ellen Werring, who hosted me in her house during this period, with all her hospitality.

I thank my lab mates, especially Erenay Dayanık, Osman Mutlu, Ozan Arkan Can, and Ömer Kırnap. We spent too much time together and eventually learned so much from each other. Thanks to them, the first years of my PhD journey became endurable. I also thank (in alphabetical order) Alper Kesen, Amir Navidfar, Berk Güler, Berkay Kullukçu, Berke Ataseven, Houman Bahmani-Jalali, Itır Bakış Doğru Yüksel, Kaan Telciler, Onuralp Karatum, Ozan Can Altıok, Volkan Aydıngül, for having fun with me. Here, Mehmet Ozan Aydın deserves a notable mention: My precious friend opened his house and prevented me from becoming homeless in İstanbul, which could be quite an adventure. Ömer also deserves a particular word since he always supported me during this journey.

I also want to thank my family for their unconditional love and support during this long period.

Last but certainly not least, I present my most profound appreciation to my significant other, my love, Laçin Kantarcı. This thesis also would not exist without her. She unconditionally provided her full support through thick and thin during this period. Spending time with her and witnessing her smile brings me joy every single time. I am blessed to have her in my life. I also want to thank her family for their hospitality and support.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABBREVIATIONS

AI	Artificial Intelligence
BERT	Bidirectional Encoder Representations
CNN	Convolutional Neural Networks
DL	Deep Learning
ILM	Image-Language Model
IoU	Intersection over Union
LIC	Language-Guided Image Colorization
LM	Language Model
LLM	Large Language Model
LSTM	Long-Short Term Memory
MLM	Masked Language Modeling
NLG	Natural Language Generation
NLI	Natural Language Inference
NLP	Natural Language Processing
QA	Question Answering
ReLU	Rectified Linear Unit
RES	Referring Expression Segmentation
ResNet	Residual Neural Networks
RNN	Recurrent Neural Networks
VidLM	Video-Language Model
VQA	Visual Question Answering

Chapter 1

INTRODUCTION

1.1 Motivation

Human language learning involves forming a deep connection between language and the world. As human beings, we learn to ground language to the world through communicating about our *experiences*, i.e., our *observations* about the world surrounding us and our *interactions* with it [Bisk et al., 2020a, Chandu et al., 2021]. In order to achieve this, humans rely on other sensory inputs for language learning, including tactile feedback, auditory input, and visual observations, which makes human language learning *multimodal* [Kiela and Clark, 2015, Thomason et al., 2016, Dessalegn and Landau, 2013].

Devising artificial intelligent (AI) systems that can reason like humans is a longstanding goal of the community, dating back to [Winograd, 1972]. Recent breakthroughs of deep learning (DL) models in computer vision [Krizhevsky et al., 2012, Girshick et al., 2014, Goodfellow et al., 2014] and natural language processing (NLP) [Sutskever et al., 2014, Gillick et al., 2016, See et al., 2017] fields enabled the community to advance on the multimodal tasks connecting vision and language (V&L). In this direction, people pushed the field forward by working on V&L tasks such as image captioning [Vinyals et al., 2015, Bernardi et al., 2016], visual question answering [Zhu et al., 2016a, Goyal et al., 2017a], and visual dialog [Das et al., 2017a, Das et al., 2017b]. This generation of V&L models implemented a hybrid approach using convolutional neural networks (CNNs) [LeCun et al., 1989, Simonyan and Zisserman, 2014, He et al., 2016a] and recurrent neural networks (RNNs) [Rumelhart and McClelland, 1987, Hochreiter and Schmidhuber, 1997a, Chung et al.,

2014]. Later, experts introduced the *attention* mechanism to DL models inspired by human cognition [Bahdanau et al., 2014, Xu et al., 2015, Gregor et al., 2015]. The attention mechanism benefited in many tasks by significantly improving the baselines. This led the community to explore more about attention, and we experienced another breakthrough in NLP due to the invention of *transformer* models [Vaswani et al., 2017].

Transformers heavily rely on attention mechanisms. In addition to cross-attention mechanism commonly used by preceding sequence-to-sequence encoder-decoder RNNs[1], transformers introduced a novel attention mechanism: self-attention. The implemented self-attention mechanism provided two crucial benefits which are not allowed by RNNs. First, it allowed language models (LM) to process long-term dependencies more efficiently. Second, self-attention made LM training more concurrent, i.e. parallel, since this mechanism enables processing the entire sequence at single time.

These benefits resulted in the emergence of different types of large-scale pretrained transformers, including encoder-only [Devlin et al., 2019, Lan et al., 2020], decoder-only [Radford et al., 2019, Zhang et al., 2022] and encoder-decoder [Lewis et al., 2020, Raffel et al., 2020], where these models outperformed RNNs by a large margin in well-known NLP problems such as machine translation, natural language inference (NLI) and question answering [Bojar et al., 2014, Wang et al., 2018, Wang et al., 2019a, Fan et al., 2019]. Subsequent to the successes of transformers in NLP, the V&L community shifted its attention towards this direction as well [Tan and Bansal, 2019a, Lu et al., 2019]. These first-generation V&L transformers employed a hybrid approach: visual features of the detected objects and textual features fused into an additional multimodal transformer being trained from scratch. These developments also pushed the vision community, resulting in the emergence of vision transformers (ViT) [Dosovitskiy et al., 2021, Ranftl et al., 2021]. Thus, newer V&L transformers

[1]In NLP, encoder-decoder sequence-to-sequence models consist of two main components: the encoder, processing the input/source language, and the decoder producing the output/target language. The cross-attention allows a direct information flow from the encoder to the decoder.

started to consist of transformers purely [Radford et al., 2021, Ramesh et al., 2021, Li et al., 2021a, Li et al., 2022b, Li et al., 2023c], leaving region-based CNNs behind. All of these immense advances of the transformers in these domains eventually allowed the AI community to move towards more challenging directions, most notably problems that require temporal reasoning and/or commonsense reasoning [Yi et al., 2020, Wu et al., 2021, Xiao et al., 2021, Liang et al., 2022, Cheng et al., 2023].

1.2 Scope of the Thesis

In this thesis, we focus on V&L problems that require either commonsense reasoning, temporal reasoning or both at the same time.

Temporal reasoning is the ability to reason over time dimension, which opens up additional challenges. In the context of vision-language learning, this means going beyond static images, i.e., videos. Video understanding requires recognizing actions and localizing events in the temporal axis, in addition to image understanding skills like scene detection and object localization. Recent studies [Buch et al., 2022, Lei et al., 2023] revealed that V&L models processing images as visual input, i.e. image-language models, could outperform video-language models in commonly used text-to-video retrieval and video question answering benchmarks [Xu et al., 2016, Xu et al., 2017, Hendricks et al., 2017, Yu et al., 2019]. In this thesis, we focus on creating novel video-language evaluation benchmarks to push video-language models to the next stage by prioritizing temporality.

Commonsense knowledge is implicit general knowledge about the world, which is gained through experience. Capturing commonsense knowledge is necessary to make judgments about the world surrounding us. Though, it is not enough: one needs to take the particular context also into account to excel in **commonsense reasoning** [Sap et al., 2020, Chandu et al., 2021]. For instance, an umbrella is *usually* used to prevent getting wet in rainy weathers. However, people can also use umbrella to avoid harms of the sunlight in hot weathers. In this example, this particular context is the weather, which affects our judgment about the purpose of using or carrying an umbrella. Commonsense reasoning can be split into different categories

such as social commonsense reasoning, numerical commonsense reasoning, abductive commonsense reasoning, temporal commonsense reasoning, physical commonsense reasoning and so on [Sap et al., 2019, Lin et al., 2020a, Liang et al., 2022, Zhou et al., 2021, Zellers et al., 2021a]. In this thesis, our main focus is the V&L problems that require understanding the implicit knowledge about the physical world.

1.3 Thesis Contributions

This thesis comprises four distinct studies that connect language and vision by exploring various aspects of temporal and commonsense reasoning. Here, we summarize our contributions.

Before advancing to problems requiring temporal and commonsense reasoning, we first shift our attention to *localization* stage: We experiment with a model [Can, 2021] that enables systematic evaluation of how language-conditioning should affect the bottom-up and the top-down visual processing branches. We perform experiments on two dense prediction tasks: referring expression segmentation and language-guided image colorization. Our findings on both tasks highlight the importance of conditioning the bottom-up branch on language to ground visual concepts like colors and object categories.

The second study focuses on *temporal reasoning* and *physical commonsense reasoning*: we investigate whether the existing video-language models thrive in answering questions about complex dynamic scenes. We choose the CRAFT as our test bed, which is a video question answering dataset that requires causal reasoning about physical forces and object interactions. We show that the state-of-the-art video-language models at that point in time fall behind human performance by a large margin. In addition, we experiment with a text-only oracle baseline that processes the descriptions of the causal scene graphs, i.e. the structured representation of the events taking place in the simulations. This oracle baseline outperforms all other models by achieving a near-human performance. This shows that the existing video-language models are not yet proficient at detecting sequences of events taking place in videos.

In the third study, we develop a zero-shot structured benchmark to evaluate the language understanding abilities of pretrained large-scale video-language models. This benchmark focuses on processing everyday actions in the temporal dimension. Our foiling benchmark offers two main tests: (i) repetitive action counting and (ii) rare action recognition. Our experiments reveal that in terms of temporal reasoning, the current video-language models are no better than the vision-language models, which take static images as input. We also propose *proficiency tests*, which are easier than the main tests. We only consider an example correct if a model correctly excels in both main and proficiency tests: Incorporating proficiency tests leads to significant performance decreases, suggesting that many correct predictions by video-language models can be accidental or spurious.

In the last project, we work on a figurative language understanding problem called euphemism detection. Euphemisms tone down expressions about sensitive or unpleasant issues like addiction and death. The ambiguous nature of euphemistic terms makes it challenging to detect their actual meaning within a context, where commonsense knowledge and reasoning are necessities. We propose a two-stage system for this particular problem. In the first stage, we seek to mitigate this ambiguity by incorporating literal descriptions, which yields remarkable performance improvement. In the second stage, we integrate visual supervision into our system using visual imageries, two sets of images generated by a text-to-image model, taking terms and descriptions as input. We show that in this low-resource scenario, visual supervision is also beneficial in detecting euphemisms.

1.4 Thesis Structure

In this section, we describe the thesis structure. The rest of this thesis is organized as described below,

- Chapter 2 contains our analysis on two localization tasks.
- Chapter 3 introduces the CRAFT dataset and presents the experimental analysis performed.

- We describe our zero-shot video-language model evaluation benchmark in Chapter 4.

- Chapter 5 provides the details of our proposed figurative language understanding methodology.

- Chapter 6 concludes this thesis.

Each chapter starts with an introduction section. We describe our methodology in the next section(s), followed by the experimental analysis. We then briefly review the relevant literature after presenting the experimental results. We conclude each chapter by listing our contributions and the limitations of our approach.

Disclaimer. I choose to use the personal pronoun *we* over *I* throughout this thesis.

Chapter 2

LOCALIZATION: CONDITIONING BOTTOM-UP AND TOP-DOWN VISUAL PROCESSING ON LANGUAGE

2.1 Introduction

This chapter focuses on the localization stage [1]. Recent studies in cognitive science [Boutonnet and Lupyan, 2015, Lupyan and Clark, 2015] suggest that language affects low-level visual processing, in addition to top-down attention. Motivated by these findings, we investigate how language-conditioning in visual processing branches affects grounding language to vision in vision-language learning. To do so, we use the U-Net-based model proposed in [Can, 2021], since it allows us to condition language on either the top-down visual processing branch, the bottom-up visual branch, or both branches at the same time. We extend the proposed model [Can, 2021] by making it adaptable to the other vision-language (V&L) tasks involving dense prediction. We perform comprehensive experiments on two V&L localization tasks: referring expression segmentation and language-guided image colorization.

Referring Expression Segmentation (RES) is a vision-language task involving dense prediction, where the aim is to identify and segment objects in an image, described in natural language. The natural language descriptions of the objects in images can include a variety of information, such as their visual attributes (e.g. *colors*), spatial relationships (e.g. *next to*), actions (e.g. *sitting*), and interactions with other objects (e.g. "the dog that is chasing ball").

In *language-guided image colorization* (LIC) task, given a grayscale image and a

[1] This work appeared a result of collaboration with Ozan Arkan Can. See [Can, 2021] for his contributions. In this chapter, I focused on only my individual contributions. See [Kesen et al., 2022a] for the complete report.

description, the aim is to predict pixel color values. The absence of color information in the input images makes this problem interesting to experiment with because color words do not help in conditioning the bottom-up branch when the input image is grayscale.

We find that conditioning both branches leads to better results, achieving competitive performance on both tasks. Our experiments suggest that conditioning the bottom-up branch on language is important to ground low-level visual information. On RES, we find that modulating only the bottom-up branch performs significantly better than modulating only the top-down branch especially when color-dependent language is present in the input. Our findings on LIC show that when color information absent in input images, the bottom-up baseline naturally fails to predict and manipulate colors of target objects specified by input language. That said, conditioning the bottom-up branch still improves the colorization quality by helping our model to accurately segment and colorize the target objects as a whole.

The rest of this chapter is structured as follows: We summarize related work and compare it to our approach in Section 2.4. We describe our model in detail in Section 2.2. We share the details of our experiments in Section 2.3. Section 2.5 summarizes our contributions.

2.2 Model

Here, we describe the evaluated model in detail. Figure 2.1 illustrates this model, originally proposed by [Can, 2021]. First, the model extracts a tensor of low-level visual features using a pre-trained convolutional neural network (CNN) and processes the given natural language expression using an LSTM [Hochreiter and Schmidhuber, 1997a]. A multi-modal encoder takes these low-level visual features and language representation, and then it generates feature maps through a contracting and expanding path similar to U-Net [Ronneberger et al., 2015]. This particular architecture modulates both of the contracting and expanding paths using language-conditioned convolutional filters generated from the given expression. Eventually, a task-specific output head processes the final feature map of the multi-modal encoder

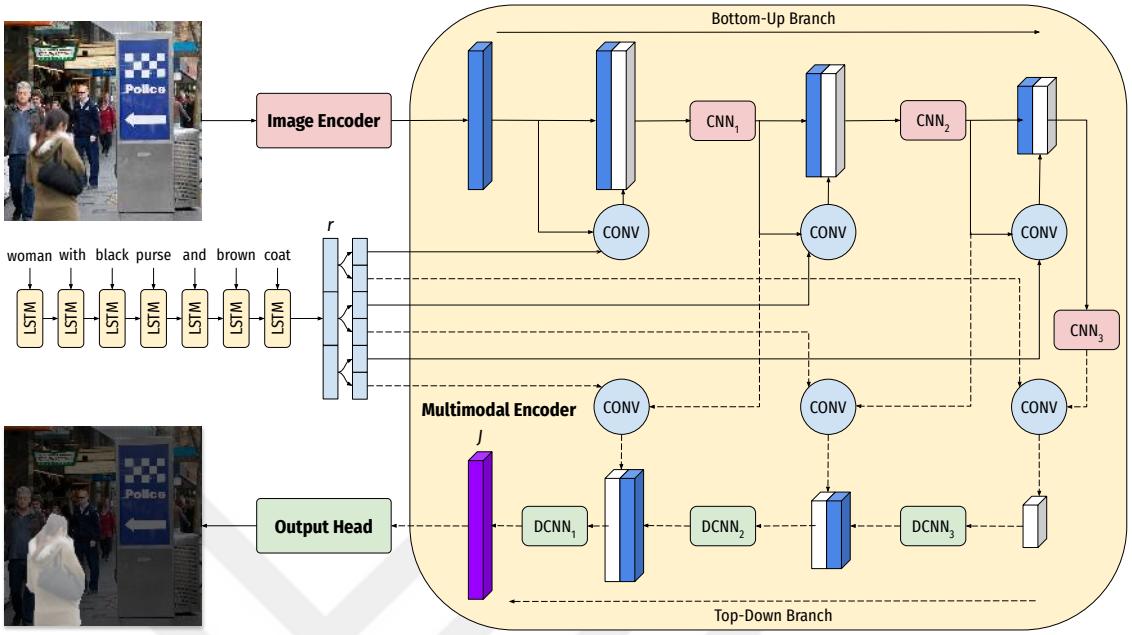


Figure 2.1: Illustration of the evaluated model on the task of referring expression segmentation. A pre-trained image encoder extracts low-level visual features. An LSTM network process the input language, and then the model chops up the final hidden state r into pieces to generate language-conditional filters. Language-conditional filters and low-level visual features are fed into the multi-modal encoder. Each layer i of the bottom-up/contracting branch convolves the previous feature map F_{i-1} with language-conditional filters, concatenates output with previous feature map, and then a downsampling module CNN_i processes this concatenated feature map. Each layer i of the top-down/expanding branch convolves the bottom-up feature map obtained through the skip-connection, concatenates it with the previous top-down feature map H_{i+1}, and then an upsampling module DCNN_i processes this concatenated feature map. Finally, an output head takes the final feature map J, and performs dense prediction.

to produce the final prediction map which has the same size with the input image. It is important to emphasize that the other works either have a language-guided top-down or a language-conditional bottom-up visual processing branch, and this particular architecture has both. As will be discussed in the next section, our

experiments show that how language-conditioning affects these visual processing branches.

2.2.1 Low-level Image Features

Given an input image I, [Can, 2021] extracts the low-level visual features I_0 by using a pre-trained convolutional network. As different from [Can, 2021], we use a more recent CNN backbone on the RES task, which is the backbone of DeepLab-v3+ semantic segmentation architecture [Chen et al., 2018b]. In the LIC task, we use ResNet-101 pre-trained on ImageNet [He et al., 2016b, Deng et al., 2009a], to make a fair comparison with previous work. On the task of referring expression segmentation, we also employ 8-D location features, following the previous work [Liu et al., 2017, Ye et al., 2019].

2.2.2 Language Representation

To process a language input $S = [w_1, w_2, ..., w_n]$, we follow the exactly same procedure with [Can, 2021]: each word in the language input w_i is mapped to a 300-dimensional GloVe embedding [Pennington et al., 2014a]. An LSTM takes these word embeddings, and processes them. The final hidden state of this LSTM is integrated as the language representation. Later on, we split this language representation into pieces to generate language-conditional filters.

2.2.3 Multi-modal Encoder

The multi-modal encoder of the evaluated model [Can, 2021, Kesen et al., 2022a] takes the image representation I_0 and the language representation, and produces a multi-modal feature map representing both modalities. This particular multi-modal encoder extends U-Net [Ronneberger et al., 2015] by conditioning both the top-down and the bottom-up visual processing branches on language.

In the bottom-up branch, the evaluated model [Can, 2021] applies m convolutional blocks to the image representation I_0. Each convolutional block, CNN_i, processes these two inputs: (i) the feature map (F_{i-1}) generated by the previous block CNN_{i-1}

and (ii) another feature map, which produced by convolution of F_{i-1} using language-conditional dynamic filters K_i^F. Each convolutional block concatenates these two feature maps, and then produces an output feature map (F_i). Each CNN_i block consists of three consecutive operations, which are (i) a convolution layer, (ii) a batch normalization layer [Ioffe and Szegedy, 2015] and (iii) a ReLU activation function [Maas et al., 2013]. Each convolution layer has 5×5 filters with $stride = 2$ and $padding = 2$, and each one reduces the spatial resolution by half. The number of channels is fixed throughout the multi-modal encoder.

Similar to [Misra et al., 2018], [Can, 2021] also splits the textual representation r to m equal parts (t_i), and then use each t_i to generate a language-conditional filter for ith bottom-up layer (K_i^F):

$$K_i^F = \text{AFFINE}_i^F(t_i) \tag{2.1}$$

Each AFFINE_i^F module performs an affine transformation on the input, and then normalization and reshaping are applied to its output to use as language-conditional convolutional filters. A convolutional layer processes the previous feature map using these language-conditional filters and produces the multi-modal feature map for the next layer:

$$G_i^F = \text{CONVOLVE}(K_i^F, F_{i-1}) \tag{2.2}$$

These language-conditioned feature map (G_i^F) for ith bottom-up layer and the previously generated feature map (F_{i-1}) are concatenated and then fed into the subsequent convolutional block CNN_{i+1}.

In the top-down branch, [Can, 2021] generates m feature maps starting from the final output of the bottom-up visual processing branch as:

$$G_i^H = \text{CONVOLVE}(K_i^H, F_i) \tag{2.3}$$

$$H_m = \text{DCNN}_i(G_m^H) \tag{2.4}$$

$$H_i = \text{DCNN}_i(G_i^H \oplus H_{i-1}) \tag{2.5}$$

Similar to the bottom-up branch, G_i^H is the modulated feature map with language-conditional filters defined as:

$$K_i^H = \text{AFFINE}_i^H(t_i) \tag{2.6}$$

where AFFINE_i^H, again, an affine transformation on the input, and then normalization and reshaping operations are applied to its output to create language-conditional convolutional filters for the ith layer of the top-down branch. The tested model implements a convolution operation over the feature maps from the contracting branch (F_i) using these language-conditional filters (K_i^H). Each upsampling block DCNN_i takes the concatenation ($\oplus$) of the language-conditioned features and the feature map (H_i) produced by the previous block. The first block operates only on the final output of the bottom-up branch. Each DCNN_i implements a deconvolution layer followed by a batch normalization and ReLU activation function. Final output H_1 becomes our joint feature map J representing the input image / language pair. Each deconvolution layer uses 5×5 filters with $stride = 2$ and $padding = 2$, where each one doubles the spatial resolution, and they all have the same number of output channels.

2.2.4 Output Heads

We extend the evaluated model [Can, 2021] by making it as a generic solution which can be used to solve V&L problems involving dense prediction. In this direction, we adapt it to two different dense prediction tasks by varying the output head: referring expression segmentation and language-guided image colorization.

Segmentation. In the referring expression segmentation problem, the goal is to generate segmentation mask for a given input image and language pair. We follow the exactly same procedure with [Can, 2021] to implement this output head. Subsequent to generating the joint feature map J, [Can, 2021] applies a series of layers (D_1, D_2, ..., D_m) to map J to the exact input resolution. Like the upsampling blocks of the multi-modal encoder, each D_k implements a deconvolution layer followed by batch normalization and ReLU activation operations. The number of channels is fixed

Method	UNC			UNC+			G-Ref			ReferIt
	val	testA	testB	val	testA	testB	val (G)	val (U)	test (U)	test
CMSA [Ye et al., 2019]	58.32	60.61	55.09	43.76	47.60	37.89	39.98	-	-	63.80
STEP [Chen et al., 2019a]	60.04	63.46	57.97	48.19	52.33	40.41	46.40	-	-	64.13
BRINet [Hu et al., 2020]	61.35	63.37	59.57	48.57	52.87	42.13	48.04	-	-	63.46
CMPC [Huang et al., 2020]	61.36	64.53	59.64	49.56	53.44	43.23	49.05	-	-	65.53
LSCM [Hui et al., 2020]	61.47	64.99	59.55	49.34	53.12	43.50	48.05	-	-	66.57
EFN [Feng et al., 2021]	62.76	65.69	59.67	51.50	55.24	43.01	**51.93**	-	-	**66.70**
BUSNet [Yang et al., 2021]	63.27	66.41	**61.39**	**51.76**	**56.87**	**44.13**	50.56	-	-	-
The evaluated model	**64.63**	**67.76**	61.03	**51.76**	56.77	43.80	50.88	**52.12**	**52.94**	66.01
MCN† [Luo et al., 2020b]	62.44	64.20	59.71	50.62	54.99	44.69	-	49.22	49.40	-
CGAN† [Luo et al., 2020a]	64.86	68.04	62.07	51.03	55.51	44.06	46.54	51.01	51.69	-
LTS† [Jing et al., 2021]	65.43	67.76	63.08	54.21	58.32	48.02	-	54.40	54.25	-
VLT† [Ding et al., 2021a]	65.65	68.29	62.73	**55.50**	59.20	**49.36**	49.76	52.99	**56.65**	-
The evaluated model†	**67.01**	**69.63**	**63.45**	55.34	**60.72**	47.11	**53.51**	**55.09**	55.31	**57.09**

Table 2.1: Comparison with the previous work by using the overall IoU metric. † denotes the corresponding method uses the mean IoU metric. "-" indicates that the model has not been evaluated on that dataset.

throughout this output head except the last layer D_m, which produces the mask prediction. Note that, we omit the batch normalization and the ReLU activation for the final layer D_m, instead we apply a sigmoid function to turn the final features into probabilities. Given these probabilities and ground-truth mask, we train our network by using binary cross entropy loss.

Colorization. In the language-guided image colorization task, the goal is to predict pixel color values for given input image with the guidance of language input. A convolutional layer by with 3×3 filters generates class scores for each spatial location of J. We apply bilinear upsampling to these predicted scores to match input image size. Given predicted scores and ground-truth color classes, we train the model by using a weighted cross entropy loss. To create compound LAB color classes and their weights, we follow the exactly same process with [Manjunatha et al., 2018, Zhang et al., 2016b].

2.3 Experimental Analysis

This section contains the details of our experiments on referring expression segmentation (Section 2.3.1) and language-guided image colorization (Section 2.3.2) tasks.

2.3.1 Referring Expression Segmentation

Datasets. We evaluate the model on ReferIt (130.5k expressions, 19.9k images) [Kazemzadeh et al., 2014], UNC (142k expressions, 20k images), UNC+ (141.5k expressions, 20k images) [Yu et al., 2016b] and Google-Ref (G-Ref) (104.5k expressions, 26.7k images) [Mao et al., 2016] datasets. Unlike UNC, location-specific expressions are excluded in UNC+ through enforcing annotators to describe objects by their appearance. ReferIt, UNC, UNC+ datasets are collected through a two-player game and have short expressions (avg. 4 words). G-Ref have longer and richer expressions, its expressions are collected from Amazon Mechanical Turk instead of a two-player game. G-Ref does not contain a test split, and [Nagaraja et al., 2016] extends it by having separate splits for validation and test, which are denoted as val (U) and test (U).

Evaluation Metrics. We use intersection-over-union (IoU) and $p@X$ as evaluation metrics following prior work. IoU is calculated as the intersection of the predicted and the ground truth segmentation masks, divided by their union. There are two different ways to calculate IoU: the overall IoU calculates the total intersection over total union score throughout the entire dataset and the mean IoU calculates the mean of IoU scores of each individual example. For a fair comparison, we use both IoU metrics. The other metric, $p@X$, computes the percentage of examples with an IoU score above a certain threshold, X.

Implementation Details. We try to choose the same hyper-parameters with the original implementation [Can, 2021]. The maximum length of input expressions is limited to 20. Image size is set to 512×512 and 640×640 for training and inference phase respectively. As different from [Can, 2021], we use the first four layers of

DeepLabv3+ with ResNet-101 backbone, pre-trained on COCO dataset by excluding images appear on the validation and the test sets of UNC, UNC+ and G-Ref datasets similar to previous work [Luo et al., 2020b, Ding et al., 2021a, Yu et al., 2018]. Thus, our low-level visual feature map I has the size of $64 \times 64 \times 64 \times 1032$ in training, and $80 \times 80 \times 1032$ in inference phase, both including 8-D location features. Each convolutional layers uses $5x5$ filters, where stride is equal to 2. We fix the number of channels of the feature maps by setting it to 512, which also applies for the output head of the segmentation model. The depth is 4 in the multimodal encoder part of the network. For the bottom-up-only baseline, we used grouped convolution in the bottom-up branch to prevent linguistic information leakage to the top-down visual branch. We apply dropout regularization [Srivastava et al., 2014] to language representation r with 0.2 probability. We use Adam optimizer [Kingma and Ba, 2014a] with default parameters. We freeze the DeepLab-v3+ ResNet-101 weights. There are 32 examples in each minibatch. We train the reimplemented model for 20 epochs on a Tesla V100 GPU with mixed precision, where one epoch takes at most two hours depending on the dataset.

Quantitative Results. Table 2.1 compares the re-implementation of the evaluated model with previous methods. Bold faces highlight the highest achieved scores. We evaluate the model using both IoU metrics for a fair evaluation. Table 2.2 compares the evaluated model with the state-of-the-art using $p@X$ metric. The performance difference between the evaluated model and the others increases as the threshold increases. This demonstrates that the evaluated model is more proficient in segmenting the referred objects, including smaller ones.

Qualitative Results. We visualize some of the segmentation predictions of the evaluated model to gain better insights about it. Figure 2.2 shows some correct predictions. Our findings are similar to [Can, 2021]. The evaluated model is able to sufficiently process superlative adjectives (e.g., *taller*) and spatial relations (e.g., *near to*). The evaluated model can also learn the concepts specific to the domain, such as the concept of a "catcher", which is present in the dataset. Lastly, we can also observe that the model has the ability to identify certain non-dynamic actions

Method	*p@0.5*	*p@0.6*	*p@0.7*	*p@0.8*	*p@0.9*
CMSA	66.44	59.70	50.77	35.52	10.96
STEP	70.15	63.37	53.15	36.53	10.45
BRINet	71.83	65.05	55.64	39.36	11.21
LSCM	70.84	63.82	53.67	38.69	12.06
EFN	73.95	69.58	62.59	49.61	20.63
MCN	76.60	70.33	58.39	33.68	5.26
LTS	75.16	69.51	60.74	45.17	14.41
The Model	**76.67**	**71.77**	**64.76**	**51.69**	**22.73**

Table 2.2: Results achieved on the val set of UNC dataset using *p@X* metrics.

Figure 2.2: A selection of the predictions of the evaluated model on the UNC validation set.

(e.g., *sitting*).

Figure 2.3 presents a couple of failure cases on the UNC test split. Our findings, again, are very similar to the original work [Can, 2021]. Examples in (a) show that the model cannot handle typos. The evaluated model segments the correct objects for

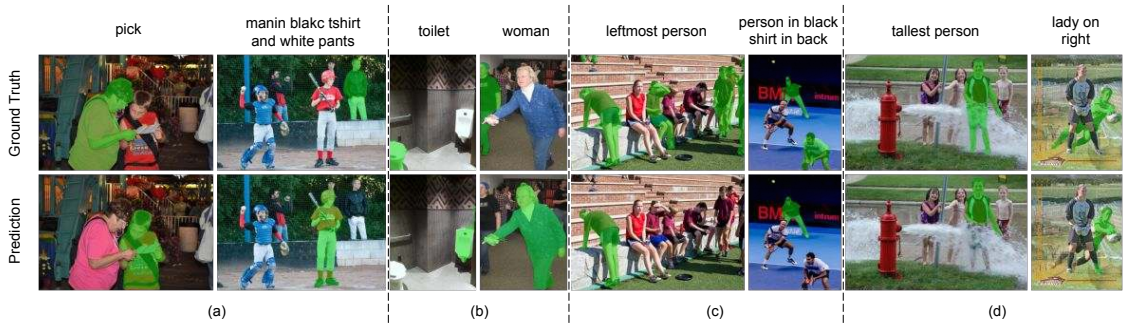


Figure 2.3: Some incorrect predictions on the UNC validation set. Each group (a-d) illustrates a different pattern. The first row shows the ground truth mask, and the second mask is the model prediction.

Method	Backbone	*IoU*
Top-down Baseline	ResNet-50	58.06
Bottom-up Baseline	ResNet-50	60.74
The Evaluated Model	ResNet-50	63.59
The Evaluated Model	ResNet-101	**64.63**

Table 2.3: Ablation study on the UNC validation split using overall *IoU* metric.

these two examples when the typos are fixed (e.g., *pink* instead of *pick*). Examples in (b) demonstrate that some expressions are ambiguous, where the expression could refer to multiple objects. In this case, the model seems to segment the most salient object. Some annotations possess incorrect or incomplete ground-truth segmentation masks (c). Finally, some cases (d) are challenging to segment simply due to the lack of light or occlusions.

Ablation Study. We test three separate models, the top-down baseline, the bottom-up baseline, and the complete model, to reveal the impact of language conditioning in expanding and contracting visual branches. While the bottom-up baseline modulates language in the bottom-up branch only, the top-down baseline only modulates language in the top-down branch. The complete model conditions language on both branches. Table 2.3 reports the results. The bottom-up baseline surpasses the

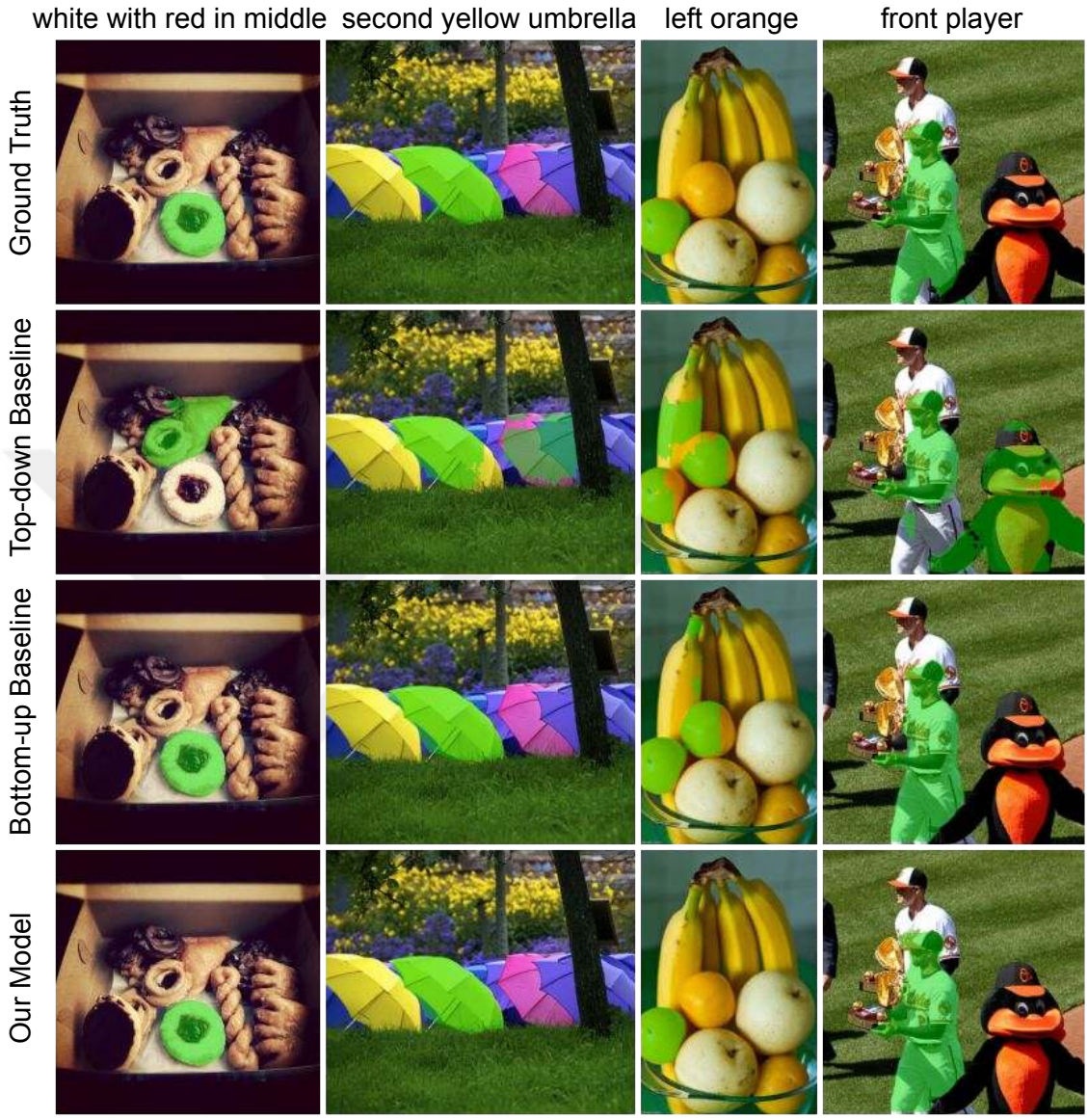


Figure 2.4: Comparison of different architectural setups on the UNC test samples.

top-down one with ≈2.7 IoU gain. Nonetheless, the best results are achieved by conditioning both branches on language, where the complete model improves the bottom-up baseline with ≈2.85 IoU score.

Figure 2.4 visualizes the predictions of the different models on the same examples. The bottom-up baseline performs better when the description has color information as we show in the first three examples. The top-down-only baseline also fails to

Method	+C	-C	+IN	+IN,-C	+JJ*	+JJ*,-C
Top-down Baseline	52.59	59.59	54.84	55.66	57.66	61.16
Bottom-up Baseline	60.40	60.02	56.05	55.49	61.18	61.86
The Evaluated Model	62.98	63.57	59.60	59.27	64.45	65.56

Table 2.4: IoU performance of different setups depending on the input expression category for the UNC test splits.

detect object categories in some cases, and segments additional unwanted objects with similar category or appearance (e.g. *banana* vs. *orange*). Overall, the evaluated model which conditions both visual branches on language gives the best results.

Language-oriented Analysis. To analyze the effect of language on model performance, we divided UNC test splits into subsets depending on the different types of words (e.g. colors) and phrases (e.g. noun phrases with multiple adjectives) included in input expressions. Table 2.4 shows us the results of different models on these subsets. The first column stands for models, and the rest stand for different input expression categories. We exclude the categories which do not contribute to our analysis. We use DeepLab-v3+ ResNet-50 as visual backbone in each method. The notation of the categories are similar to Part of Speech (POS) tags [Marcus et al., 1993], where we denote prepositions with IN, examples with adjectives with JJ*, and colors with C. Preceding plus and minus signs stand for inclusion and exclusion. For instance, +IN,-C column stands for the subset where each expression contains at least one preposition without any color words. Color words (e.g. red, darker) has the most impact on the performance in comparison to other types of words and phrases. The evaluated model and the bottom-up baseline performs significantly better than top-down baseline on the subset that includes colors. In the opposite case, where input expressions with colors are excluded, the top-down baseline has performance similar to the bottom-up baseline, and our final model outperforms both single branch models. Since colors can be seen as low-level sensory information, low performance in the absence of the bottom-up branch can be expected. This demonstrates the

importance of conditioning the bottom-up visual branch on language to capture low-level visual concepts.

2.3.2 Language-guided Image Colorization

Datasets. Following the prior work [Manjunatha et al., 2018], we use a modified version of COCO dataset [Lin et al., 2014b] where the descriptions that do not contain color words are excluded. In this modified version, the training split has 66165, and the validation split has 32900 image / description pairs, and all images have a resolution of 224×224 pixels.

Evaluation Metrics. Following the previous work, we use pixel-level top-1 (acc@1) and top-5 accuracies (acc@5) in LAB color space, and additionally PSNR and LPIPS [Zhang et al., 2018] in RGB for evaluation. A lower score is better for LPIPS, and a higher score is better for the rest.

Implementation Details. Unless otherwise speficied, we follow the same design choices applied for the referring expression segmentation task. We set the number of language-conditional filters as 512, replace the LSTM encoder with a BiLSTM encoder, and we use the first two layers of ResNet-101 trained on ImageNet as image encoder to have a similar model capacity and make a fair comparison with the previous work [Manjunatha et al., 2018]. We set input image width and height to 224 in both training and validation. Thus, the low-level visual feature map has the size of $28 \times 28 \times 512$, and we don't use location features. Additionally, in our experimental analysis, we consider the same design choices with previous work [Manjunatha et al., 2018, Zhang et al., 2016b]. Specifically, we use LAB color space, and the evaluated model predicts *ab* color values for all the pixels of the input image. We perform the class re-balancing procedure to obtain class weights for weighted cross entropy objective. We use 313 *ab* classes present in ImageNet dataset, and encode *ab* color values to classes by assigning them to their nearest neighbors. We use input images with a size of 224×224, and output target images with a size of 56×56 which is same with the previous work.

Quantitative Results. We present the quantitative performance of the evaluated

Method	acc@1	acc@5	PSNR	LPIPS
FiLMed ResNet	23.70	60.50	-	-
FiLMed ResNet (ours)	20.22	49.57	20.89	0.1280
Top-down Baseline	22.83	51.85	21.29	**0.1226**
Bottom-up Baseline	21.85	51.34	20.98	0.1448
The Evaluated Model	23.38	54.27	21.42	0.1262
The Evaluated Model w/o balancing	**33.74**	**67.83**	**22.75**	0.1250

Table 2.5: Colorization results on the modified COCO validation split.

model in Table 2.5, and compare it with different design choices and previous work. FiLMed ResNet [Manjunatha et al., 2018] uses FiLM [Perez et al., 2018] to perform language-conditional colorization. FiLMed ResNet (ours) denotes the results reproduced by the implementation provided by the authors. To show the effect of language modulation on different branches, we train 3 different models again: the top-down baseline, the bottom-up baseline and the evaluated model. We also re-train the evaluated model without class rebalancing and denote it as *The Evaluated Model w/o balancing*.

Contrary to the segmentation experiments, the top-down baseline performs better than the bottom-up baseline on the colorization task in all measures. Since, color information is absent in input images, bottom-up branch cannot encode low-level image features by modulating color-dependent language.

When we disable class rebalancing in the training phase, we observe a large improvement in acc@1 and acc@5 due to the imbalanced color distribution, where the model predicts the frequent colors exist in the backgrounds.

Qualitative Results. We visualize some of the colorization outputs of the trained models to analyze them in more detail in Figure 2.5. FiLMed ResNet (ours) can understand all colorization hints, and it can manipulate object colors with some incorrectly predicted areas. The top-down baseline also performs similar to FiLMed ResNet (ours), where both models condition only the top-down branch on language.

In this task, since the models are blind to color, the bottom-up baseline loses its effectiveness to some degree, and starts to predict the most probable colors. This can be seen on the second and the last example, where the bottom-up baseline predicts red for the stop sign and blue for the sky. Although, the bottom-up baseline performs worse in this task, modulating the bottom-up branch with language still contributes to the final model to localize and recognize the objects present in the scene. This can be seen on the last two examples, where the top-down baseline mixes colors up in some object parts (e.g. the red parts in the motorcycle). The evaluated model w/o balancing tends to predict more grayish colors (e.g. dog, sky).

Figure 2.6 highlights some of the failure cases we observed throughout the dataset. In the first two examples, the evaluated model is able to localize and recognize the target objects, but it fails to colorize them successfully by colorizing not only the targeted parts but also other parts. Models generally fail to colorize small objects since the data is imbalanced and it contains vast backgrounds and big objects frequently. The last two examples show that models fail to colorize reflective or transparent objects like glasses or water, these were also difficult in the language based segmentation task (see Figure 2.3 (d)).

2.4 Related Work

Here we briefly review prior work on referring expression comprehension, referring expression segmentation, language-guided image colorization, dynamic filters and compare them with the evaluated method.

2.4.1 Referring Expression Comprehension

In Referring Expression Comprehension (REC) problem, the goal is to locate a bounding box for the object(s) described in the input language. The proposed solutions can be divided into two categories: two-stage and one-stage methods. Two-stage methods [Mao et al., 2016, Hu et al., 2017, Nagaraja et al., 2016, Yu et al., 2018, Wang et al., 2019c, Cirik et al., 2018, Hu et al., 2017, Yu et al., 2018, Liu et al., 2019a, Yang et al., 2020a] rely on a pre-trained object detector [Ren et al.,

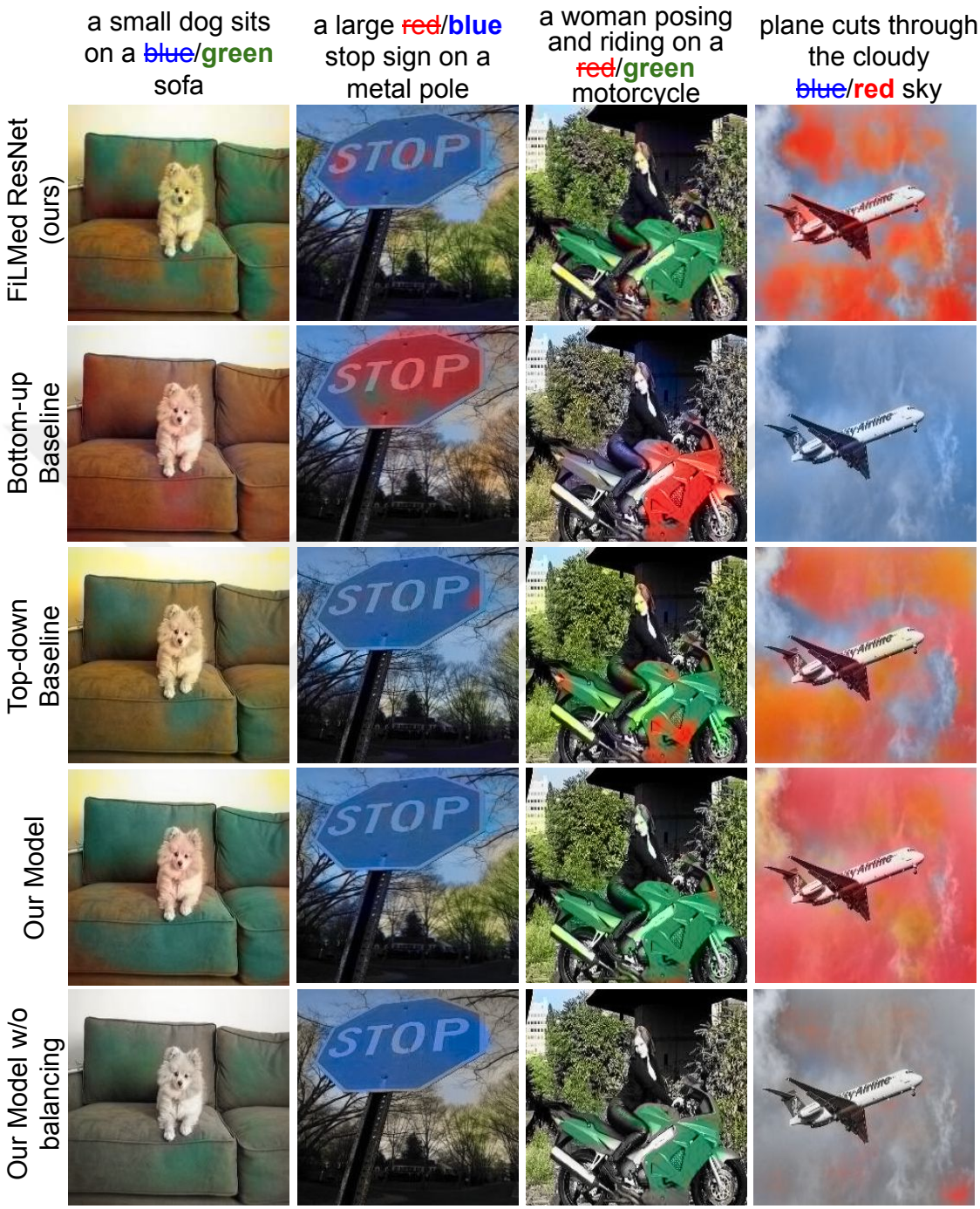


Figure 2.5: Color manipulation performance of different models on COCO validation examples. Each column focuses on a different color conversion.

2017, He et al., 2017] to generate object proposals in the first stage. In the second stage, they assign scores to the object proposals depending on how much they match

Figure 2.6: Failure cases on the language-guided image colorization task.

with input language. One-stage methods [Luo et al., 2020b, Yang et al., 2020b, Liao et al., 2020, Yang et al., 2019, Yang et al., 2020c, Deng et al., 2021] directly localize the referred objects in one step. Most of these methods condition only the top-down visual processing on language, while some fuse language with multi-level visual representations.

2.4.2 Referring Expression Segmentation

In Referring Expression Segmentation (RES) task, the aim is to generate a segmentation mask for the object(s) referred in the input language [Hu et al., 2016]. To accomplish this, multi-modal LSTMs [Liu et al., 2017, Margffoy-Tuay et al., 2018], ConvLSTMs [Shi et al., 2015, Liu et al., 2017, Chen et al., 2019a, Ye et al., 2020], word-level attention [Shi et al., 2018, Hui et al., 2020, Chen et al., 2019a, Ye et al., 2020], cross-modal attention [Ye et al., 2019, Hu et al., 2020, Huang et al., 2020, Luo et al., 2020b, Luo et al., 2020a, Jing et al., 2021], and transformers [Vaswani et al., 2017, Ding et al., 2021a] have been used. Each one of these methods modulates

only the top-down branch with language. As one exception, EFN [Feng et al., 2021] conditions the bottom-up branch on language, but does not modulate the top-down branch with language.

2.4.3 Language-guided Image Colorization

In Language-guided Image Colorization (LIC) task, the aim is to predict colors for all the pixels of a given input grayscale image based on input descriptive text. Specifically, [Manjunatha et al., 2018] inserts extra Feature-wise Linear Modulation (FiLM) layers [Perez et al., 2018] into a pre-trained ResNet to predict color values in LAB color space. Multi-modal LSTMs [Liu et al., 2017, Zou et al., 2019] and generative adversarial networks [Goodfellow et al., 2014, Bahng et al., 2018, Chen et al., 2018a] are also used in this context to colorize sketches. Similar to the evaluated model, Tag2Pix [Kim et al., 2019] extends U-Net to perform colorization on line art data, but it modulates only the top-down visual processing with symbolic input using concatenation.

2.4.4 Language-conditional Parameters

Here we review methods that use input-text-dependent dynamic parameters to process visual features. To control a visual model with language, MODERN and FiLM [De Vries et al., 2017, Perez et al., 2018] used conditional batch normalization layers with language-conditioned coefficients rather than customized filters. Numerous methods [Li et al., 2017, Gao et al., 2018, Gavrilyuk et al., 2018, Margffoy-Tuay et al., 2018, Chen et al., 2019b, Misra et al., 2018] generate language-conditional dynamic filters to convolve visual features. Some RES models [Margffoy-Tuay et al., 2018, Chen et al., 2019b] also incorporate language-conditional filters into the their top-down visual processing. To map instructions to actions in virtual environments, LingUNet [Misra et al., 2018] extends U-Net by adding language-conditional filters to the top-down visual processing only. Each one of these methods conditions either top-down or bottom-up branch only.

2.4.5 Comparison

To support our main research question, the evaluated model clearly separates bottom-up and top-down visual processing. This allows us to experiment with modulating either one branch or both branches with language and evaluate their individual contributions. The majority of related work conditions only the top-down visual processing on language. Other U-Net-based methods [Misra et al., 2018, Kim et al., 2019] and most transformer models [Ding et al., 2021a, Deng et al., 2021] which implement cross-modal attention between textual and visual representations in top-down visual processing fall into this category. A few exceptions [De Vries et al., 2017, Perez et al., 2018, Feng et al., 2021] do the opposite by conditioning only the bottom-up branch. Some methods [Luo et al., 2020b, Yang et al., 2019, Huang et al., 2020] fuse language with a multi-level visual representation, which leads to good results, but this kind of fusion does not allow the evaluation of language conditioning on top-down vs. bottom-up visual processing. This circumstance also applies to the pretrained vision-language transformers, which implement a cross-modal attention mechanism between language at the word-level and image at the object-level [Lu et al., 2019, Tan and Bansal, 2019b]. The evaluated model allows language to control either or both of the top-down and bottom-up branches. We show that (i) the bottom-up conditioning is vital to ground language to low-level visual features, and (ii) conditioning both branches on language leads to the best results.

2.5 Conclusion

In this chapter, we investigated how conditioning top-down and bottom-up visual branches on language affects grounding language to vision. To achieve this, we tested a generic architecture with explicit bottom-up and top-down visual branches for V&L problems involving dense prediction. Our experiments on two different localization tasks demonstrated that conditioning both visual branches on language gives the best results. Our experiments on the referring expression segmentation task revealed that conditioning the bottom-up branch on language plays a vital role

to process color-dependent input language. The language-guided image colorization experiments demonstrated similar conclusions, the bottom-up baseline failed to colorize the target objects since the color information is absent in the input images.

Limitations. We share common failure cases in Figure 2.3 and Figure 2.6. The model performance on both tasks decreases in the presence of transparent and/or reflective objects. The model also fails to colorize small objects, mostly due to having an imbalanced color distribution. Finally, the evaluated model is limited to integrated V&L tasks involving dense prediction, and we did not perform experiments on other V&L problems.

Chapter 3

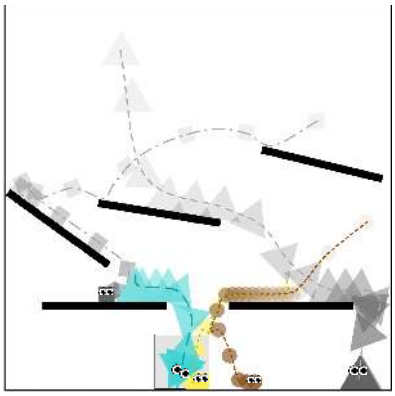
COUNTERFACTUAL AND CAUSAL REASONING ABOUT PHYSICAL INTERACTIONS

3.1 Introduction

This chapter[1] of the thesis investigates how well the existing models process physical forces acting on objects and object interactions in dynamic scenes. Artificial learning systems have shown astonishing progress recently in vision-language learning, leading the community to work on more challenging problems. One such challenging research problem is reasoning about the physical actions of objects in complex causal scenes. To achieve this, we use the CRAFT dataset [Ates et al., 2022], and analyze how well the existing video-language models process physical and causal relationships between dynamic objects in a scene through answering questions about complex dynamic scenes.

CRAFT is primarily designed to be easy to solve for humans yet difficult for video-language models. Figure 3.1 shows an overview of the CRAFT. The CRAFT dataset includes questions in three different categories called *descriptive*, *counterfactual* and *causal* questions about the physical events taking place in synthetically generated 2-dimensional complex dynamic scenes. Previous benchmarks either has limitations in their language component, or they lack the visual scene diversity and/or complexity. The CRAFT dataset does not have these shortcomings, and takes a step forward in two aspects. First, the CRAFT additionally includes questions that require *causal reasoning* in complex dynamic scenes: Answering such questions require (i) localizing objects, (ii) perceiving which physical forces acting on these objects, (iii) tracking

[1]This work appeared a result of collaboration with Tayfun Ates and others. In this chapter, I focused on only my individual contributions. See [Ates et al., 2022] for the complete report.



Q: "How many objects **fall to** the **ground**?" A: "2"
Q: "After **entering** the **basket**, does the **small yellow square** collide with other objects?" A: "True"
Descriptive Questions

Q: "How many objects **fall to** the **ground** if the **small yellow box** is removed?" A: "1"
Q: "Will the **small gray box enter** the **basket** if any of the other objects are removed?" A: "True"
Counterfactual Questions

Cause
Q: "Does the **small brown ball cause** the **big gray triangle** to fall to the **ground**?" A: "False"
Q: "Does the **small brown sphere cause** the **tiny yellow box** to **enter** the **basket**?" A: "True"

Enable
Q: "Does the **small brown sphere enable** the **small yellow box** to **enter** the **basket**?" A: "False"
Q: "How many objects does the **small gray block enable** to **enter** the **basket**?" A: "0"

Prevent
Q: "Does the **small yellow square prevent** the **tiny brown circle** from **entering** the **basket**?" A: "True"
Q: "How many objects does the **large cyan triangle prevent** from **entering** the **basket**?" A: "1"
Causal Questions

Figure 3.1: CRAFT dataset overview on a single sample scene. CRAFT includes questions in three different categories, which are descriptive questions, counterfactual questions and causal questions. To answer descriptive questions, a model needs to be proficient in spatio-temporal reasoning. Nonetheless, the other types of questions require more than spatio-temporal reasoning. Figure adapted from [Ateş, 2021, Ates et al., 2022]

the states of the objects, and (iv) comprehending causal relationship between the objects aforementioned in the question, which is also aligned with the force dynamics theory [**?**, Wolff and Barbey, 2015]. Second, the CRAFT contains questions that may require multi-step counterfactual reasoning. For example, the question *"If the large red triangle is removed, will the small blue circle fall to the ground?"* involves single counterfactual situation by directly pointing out which object is going to be removed. On the other hand, to answer a question like *"If any of the other objects are removed, will the large blue triangle end up in the basket?"*, it is necessary to consider multiple different hypothetical situations, which makes the CRAFT different from the others.

We start with the question, *"Can the existing models answer questions requiring understanding complex dynamic scenes?"*, and proceed with CRAFT as our experimental test-bed. We implement baselines in five distinct categories including simple and strong baselines. We first start with simple heuristic baselines like random prediction baselines. Text-only baselines only process the question and do not see the

visual scene at all. In the next category, we have static image-based baselines, where the models can only access to the first or last frame of the input video. Eventually, we also experiment with various video-language models that can process the input videos completely from beginning to end. In addition to these baselines, we also evaluate oracle text-only approaches in our analysis. To achieve this, we transcribe the causal scene graphs into the scene descriptions, which list the events taking place in the inputs videos chronologically.

We show that the existing models are not yet capable of achieving a good performance on CRAFT, even in descriptive type of questions. Our analysis reveals that there is a huge gap between model and human performance. Our oracle description baselines achieve outstanding performance, outperforming video-language models by a large margin, even achieving a better performance than humans. This finding shows two things: First, a model needs to detect and identify the events taking place in videos in order to excel such a challenge introduced by CRAFT. Second, the existing video-language models are unable to process the videos sufficiently. What's more, oracle text-only baselines achieve a similar performance to humans in causal and counterfactual questions, pointing out that the CRAFT benchmark might be including biases.

The rest of this chapter is structured as follows: Section 3.2 briefly describes the CRAFT benchmark. We share details regarding the models evaluated on this benchmark in Section 3.3. Section 3.4 contains the obtained results and our findings. We review relevant previous work in Section 3.5. Finally, Section 3.6 summarizes our findings, and lists some potential directions.

3.2 The CRAFT Dataset

This section gives an overview of the CRAFT dataset. CRAFT is a challenging benchmark for measuring the temporal, counterfactual, and causal reasoning capabilities of existing neural network models in dynamic scenes through visual question answering. The dataset has approximately 57K question and video pairs, which are created from 10K videos. It is split into train, validation, and test sets with a

60:20:20 ratio per video basis, meaning that video clips in the training set are not seen in the validation or test set. The CRAFT dataset comes with two different settings, an *easy setting* and a *hard setting*. They differ from each other in the way how the test split is chosen. The hard setting excludes the scene layouts that are seen during training. The easy setting does not have this constraint. In the easy setting, there are 35K, 12K, and 11K question and video pairs in the train, validation and test splits, whereas in the hard setting these numbers are 35K, 11K and 12K, respectively. We provide an example set of questions from CRAFT in Figure 3.1. CRAFT contains 7 static objects, 3 dynamic objects and 20 distinct visual scene layouts. Each dynamic object can appear in 2 different sizes and 8 different colors. Please, see the full paper for more detail [Ates et al., 2022].

Simulation Representation. We also describe the simulation representation, i.e. causal graphs, in here. A simulation instance is represented by three different data structures, *the initial state of the scene*, *the final state of the scene*, and *the causal graph of extracted events*. Figure 3.2 shows a simple causal graph example. The initial and final state of a scene refers to the information regarding the objects' static and dynamic attributes such as color, position, shape, and velocity at the start or at the end of the simulation, respectively. The final state is important as it bears causal relationships between the events of a simulation. We also use these simulation representations to create the scene descriptions, later to be used in oracle descriptions baseline. To achieve this, we simply concatenate the event descriptions chronologically. Together these information sources should have sufficient information to find the correct answers to CRAFT questions.

3.3 Baseline Models

This section shares the details of the models evaluated in this study. Overall, we try to adapt each model adapted in CLEVRER study [Yi et al., 2020]. We divide these models into four different categories based on the simulation input type: text-only models, single frame models, video-language models and oracle description models. In the following subsections, we share the details regarding these models.

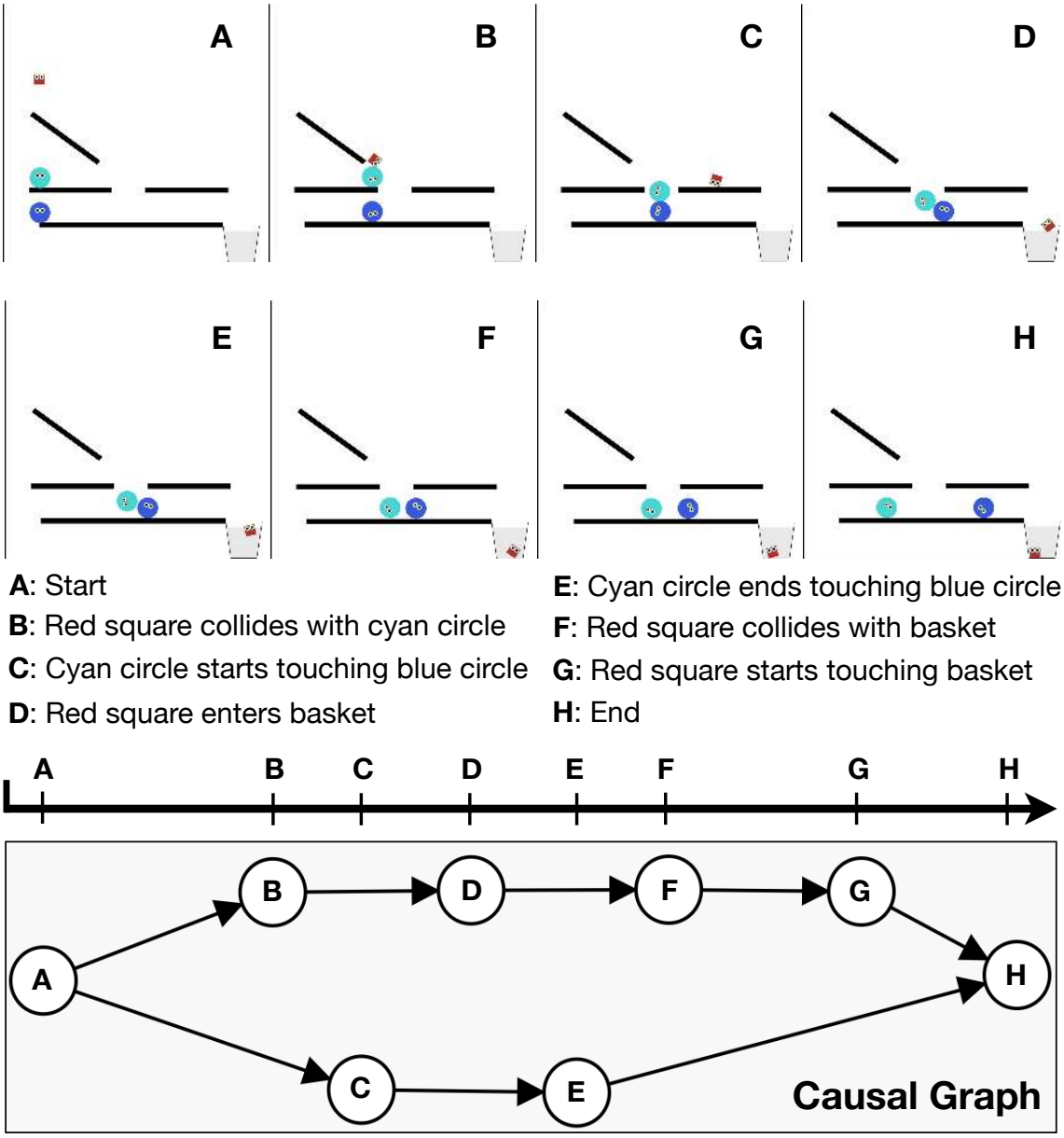


Figure 3.2: **A simple causal graph.** The causal graph is a graphical summary of the events that occur in a simulation. For the sake of simplicity, here we only include the interactions between the dynamic objects and the basket, and moreover, the scene is uncomplicated that there is no intermediate branching in the causal graph.

3.3.1 Heuristic Models

Heuristic models either perform random guesses or follow simple rules. **Random** model uniformly samples a random answer from the full answer space, whereas **Answer Type Based Random** model (AT-Random) makes random guesses based on the answer type (e.g. color, shape, boolean). **Most Frequent Answer** baseline (MFA) employs a simple heuristics and answers all the questions by using the most frequent answer in the training split. **Answer Type based Most Frequent Answer** model (AT-MFA) performs the same heuristics by taking the answer types into account similar to AT-Random baseline.

3.3.2 Text-only Models

Text-only models discard visual inputs, and do not use any visual information related to input simulations. **LSTM** model is a text-only baseline that processes the question with an LSTM [Hochreiter and Schmidhuber, 1997b], and then predicts an answer to a given question ignoring the visual input. In addition to the LSTM baseline, we experimented with **BERT** [Devlin et al., 2019] by using the `CLS` token embedding as question representation to predict answers.

3.3.3 Single-Frame Models

LSTM-CNN baseline takes both visual and textual input into account by extending the LSTM model to additionally consider the features extracted from the a pretrained ResNet-18 model [He et al., 2016b]. We evaluate both (non-temporal) single frame and video versions. In the former, each video is encoded by taking into account either the first frame or the last frame, which are referred to as **LSTM-CNN-F** and **LSTM-CNN-L**, respectively. We concatenate the extracted visual and textual features to obtain a combined representation of the video and the question pair, feeding it to a multilayer perceptron network (MLP), followed by a linear layer generating scores for the answers.

Memory, Attention, and Composition (MAC) model [Hudson and Manning,

2018] is a compositional visual reasoning model. It decomposes the reasoning task into a series of attention-guided processing steps by isolating memory and control functions from each other. The attention mechanism considers visual and textual features jointly, which leads to robust encodings of the question and the image. Similar to the LSTM-CNN baselines, **MAC-F** looks at only the first frame, and **MAC-L** only pays attention to the last frame.

3.3.4 Video-Language Models

Our first video-language model **LSTM-CNN-V** processes downsampled videos by using R3D [Tran et al., 2018] as visual feature extractor. Similar to the single frame variants LSTM-CNN-F and LSTM-CNN-L, LSTM-CNN-V also concatenates the extracted visual and textual features to obtain a joint video-question representation. Again, similar to its single frame derivatives, we use a multilayer perceptron followed by a linear projection layer to predict the scores over the entire answer vocabulary.

MAC-V baseline extends the MAC model by considering the video frames sampled from the given video as the visual input. Like LSTM-CNN-V model, MAC-V also processes videos using R3D. Unlike its non-temporal variations, MAC-F and MAC-L, where the read unit originally has spatial attention over the image, this temporal variation has a read unit that applies spatio-temporal attention over the features extracted from the entire video.

TVQA is a multi-stream state-of-the-art video question answering neural model [Lei et al., 2018]. To adapt this model to make working on the CRAFT benchmark, we only use its video stream branch and omit the answer input by generating scores for the entire answer vocabulary. In parallel with other baselines, TVQA model also extracts visual features by using ResNet-18. Different from the original implementation, our TVQA implementation uses LSTM networks with 256 units, uses a MLP network with 2 layers. Unlike the original model, we do not use GloVe word embeddings [Pennington et al., 2014b] to make a fair comparison with the remaining baseline models.

TVQA+ is another multi-stream video question answering model, which is built

upon TVQA model. In contrast to TVQA, TVQA+ uses convolutional networks as sequence encoder instead of LSTM networks, replaces GloVe word embeddings with BERT embeddings [Devlin et al., 2019], and implements a span proposal / prediction mechanism. We do not implement span proposal mechanism, and omit using BERT embeddings to compare TVQA+ with others more fairly as we disable GloVe embeddings in TVQA. Our TVQA+ implementation uses 256 hidden units in all submodules throughout the network, and it generates answer scores by feeding weighted average of fused multi-modal simulation-question representation into a linear layer.

G-SWM[2] is a recently proposed object-centric model [Lin et al., 2020b], which is originally designed for simulating possible futures in a scene consisting of multiple dynamic objects. It models each frame in a video by two different latent variables encoding object and context features. We modify G-SWM to solve the reasoning tasks in CRAFT. In particular, our version of G-SWM takes in video frames resized to 64×64 pixels and extracts an object-centric representation of the input video through object and context features. These latent codes are then combined and concatenated with the LSTM-based question representation, then an MLP followed by a linear layer processes this representation to produce answer prediction scores.

LSTM-D and **BERT-D** are *oracle* text-only baselines, which take the natural language description of the causal graph of the simulation (see Figure 3.2) as input in addition to the question. We generate these descriptions from simplified versions of the causal graphs by only considering the *Start*, *End*, *Collision* and *Enter Basket* events, and excluding those involving certain static objects (walls, platforms, ramps, and static balls) which are not mentioned in the questions. We first sort the events by their timestamps and concatenate a template-based description of each event to generate the summary. LSTM-D uses two separate LSTM networks process the question and the description, and then a linear layer predicts the answer for the input question/description pair. BERT-D extends the BERT baseline by using the descriptions as prefixes for the input questions.

[2]G-SWM is adapted by Tayfun Ates.

	Model	Easy Setting				Hard Setting			
		C	**CF**	**D**	**All**	**C**	**CF**	**D**	**All**
Heuristic	Random	5.95	5.25	5.09	5.24	5.37	4.62	5.08	4.98
	AT-Random	36.67	44.34	33.95	37.47	33.67	46.06	34.16	37.52
	MFA	32.68	43.28	23.53	30.72	30.09	43.94	23.20	29.98
	AT-MFA	49.62	47.21	37.57	42.03	49.28	47.17	36.55	41.12
Text-only	LSTM	53.04	53.14	38.29	44.69	52.51	56.24	37.25	44.52
	BERT	48.43	50.59	37.55	42.90	49.28	52.12	36.52	42.52
Single Frame	LSTM-CNN-F	53.11	55.23	44.86	49.07	48.07	48.12	35.54	40.64
	LSTM-CNN-L	54.86	55.63	43.12	48.42	49.86	54.44	38.88	44.66
	MAC-F	53.18	52.88	44.40	48.10	51.86	53.5	42.12	46.55
	MAC-L	49.97	53.08	44.54	47.83	50.21	53.8	41.46	46.05
Video	LSTM-CNN-V	54.65	61.42	48.12	53.01	51.86	54.89	41.36	46.50
	MAC-V	53.95	57.72	44.51	49.74	51.22	54.71	42.94	47.31
	TVQA	53.67	55.57	36.89	44.71	51.00	55.12	36.31	43.46
	TVQA+	54.86	60.02	40.22	48.11	51.00	55.12	39.09	45.12
	G-SWM	53.54	55.29	37.05	44.69	51.00	48.68	37.77	42.47
Oracle	LSTM-D	51.71	55.89	63.22	59.53	51.93	56.00	59.57	57.64
	BERT-D	68.44	80.05	93.41	86.20	66.33	79.34	91.30	84.90
		C		**CF**		**D**		**All**	
	Human	71.27		83.07		87.45		76.60	

Table 3.1: Performances of the tested models on the test set of the CRAFT dataset on easy and hard splits. C, CF, and D columns stand for *Causal*, *Counterfactual*, and *Descriptive* tasks, respectively. Evaluation metric is accuracy.

3.3.5 Implementation Details

Unless otherwise specified, all learnable baselines are trained with Adam optimizer [Kingma and Ba, 2014b] with default hyperparameters. We LSTM and single frame models are trained for 75 epochs with batch size of 64. All temporal baselines are trained for 30 epochs with batch size of 32. G-SWM is trained for 100 epochs using a batch size of 64 with Adam optimizer and a learning rate of 0.0001. Input videos are

downsampled at 5 frame per second (fps) and their frames are resized to 112×112 pixels. We used mixed precision strategy to train baselines more efficiently on Tesla V100 and Tesla P4 GPUs, with the exception of TVQA+ which is trained by using full precision. Training single frame models take 2 minutes, and training video models take 20-30 minutes per epoch approximately. All word embeddings have the length of 256 and are randomly initialized. Pretrained convolutional video and image encoders are jointly trained with the rest of the networks. We use negative log-likelihood loss function for all models where the models predict a distribution over the set of possible answers. All models are tuned based on their performances on the validation split. For each baseline, we report the accuracies considering the model which achieves the best performance on the validation split. We use floating point pixels values, between 0 and 1. We initialize ResNet-18 model pretrained on ImageNet 2012 dataset [Deng et al., 2009b].

3.4 Experimental Results

This section contains information about our experimental results, and findings based on these results. We first share the quantitative results and discuss these results in Section 3.4.1. In Section 3.4.2, we show a couple of qualitative examples together with the scene descriptions and the model predictions.

3.4.1 Quantitative Results

Table 3.1 presents the performances of the tested models for each question type, considering both the easy and the hard settings explained in Section 3.2. The last row of this results table stands for the human studies conducted in the original work. The evaluation metric is accuracy. As expected, the text-only models perform the worst as they completely ignore the visual information present in the videos.

Easy vs. Hard Settings

As can be seen from Table 3.1, there exists a substantial gap between the model performances in the easy and hard settings of CRAFT. Not surprisingly, this is not the case for the text-based baselines, in which it is not important whether a scene layout has been seen before during training or not. Overall, these results suggest that our tested multimodal methods are not able to generalize well to previously unseen scenes: They cannot fully detect the physical interactions and localize the events taking place in a video.

Different Question Types

The model performances vary between different question types in CRAFT. Out of the three question types, the models consistently perform poorly on the descriptive questions in that the accuracies are around 23.5%-48.12% in the easy setting and 23.2%-42.9% in the hard setting. The reason behind this could be attributed to the variety of the answers in this task as it includes questions covering both count, shape, and color of the object(s). On the other hand, the accuracies of the models on the remaining questions types are between 32.7% and 61.4% in the easy setting, and 30.1% and 56.2% in the hard setting.

Video-Language Model Performances

LSTM-CNN-V baseline does reasonably well on the easy setting, but its generalization capability on the hard setting is not that good. TVQA performs worse than the LSTM-CNN-V baseline, which shows that it is more tailor-fit to video question answering about TV clips, and its performance degrades when it does not have access to subtitles or the related concept detectors. Notably, MAC variants perform the best in the hard setting. MAC model, together with G-SWM, is a more expressive model specifically designed for compositional visual reasoning. G-SWM, however, performs poorly in our experiments, which might be because the scenes in CRAFT usually consist of many objects, thus making it harder to learn decomposing a video

into objects and background. This may be resolved by switching to a two-stage framework, in which G-SWM is pretrained first to improve its decomposition ability. For now, we left this as future work. Overall, the accuracies are not very high, indicating the shortcomings of the existing models in physical reasoning.

Oracle Description Baselines

Our oracle models, LSTM-D and BERT-D, perform better than all the tested video-language models. Interestingly, the performance of BERT-D is very close to human performance, even outperforming humans for the descriptive questions. Clearly, to excel in this task, a model must capture the interactions between the dynamic objects with each other and with the environment.

3.4.2 Qualitative Examples

Figure 3.3 provides some qualitative examples with the oracle descriptions and the model predictions. On the left side of the figure, we share some examples belonging to the descriptive question category; on the right side, we share examples from causal and counterfactual reasoning categories. The causal and counterfactual qualitative examples in the figure show the reason behind BERT-D's devastating performance. We expect that BERT-D should be unable to answer these questions accurately. However, BERT-D predicts the correct answer in the first two examples. The counterfactual situation specified in the question often does not change the answer. Additionally, one can observe that the predicted outcomes in the third and fourth examples, *yellow circle not entering basket* and *purple circle not hitting the ground*, take place in the original scenes and descriptions. These observations suggest that the CRAFT dataset contains many video-question pairs unaffected by counterfactual manipulation, indicating a high bias in this sense (75% vs. 25%). Our qualitative analysis also reveals an interesting finding: The existing video-language models *could*[3] predict colors, which are not present in the scene. This could be seen

[3]Since there are no many examples presented in here, this finding needs more investigation.

in the second descriptive question example, where LSTM-CNN-V, MAC-V, TVQA and TVQA+ models predicted the colors that are completely absent in the scene.

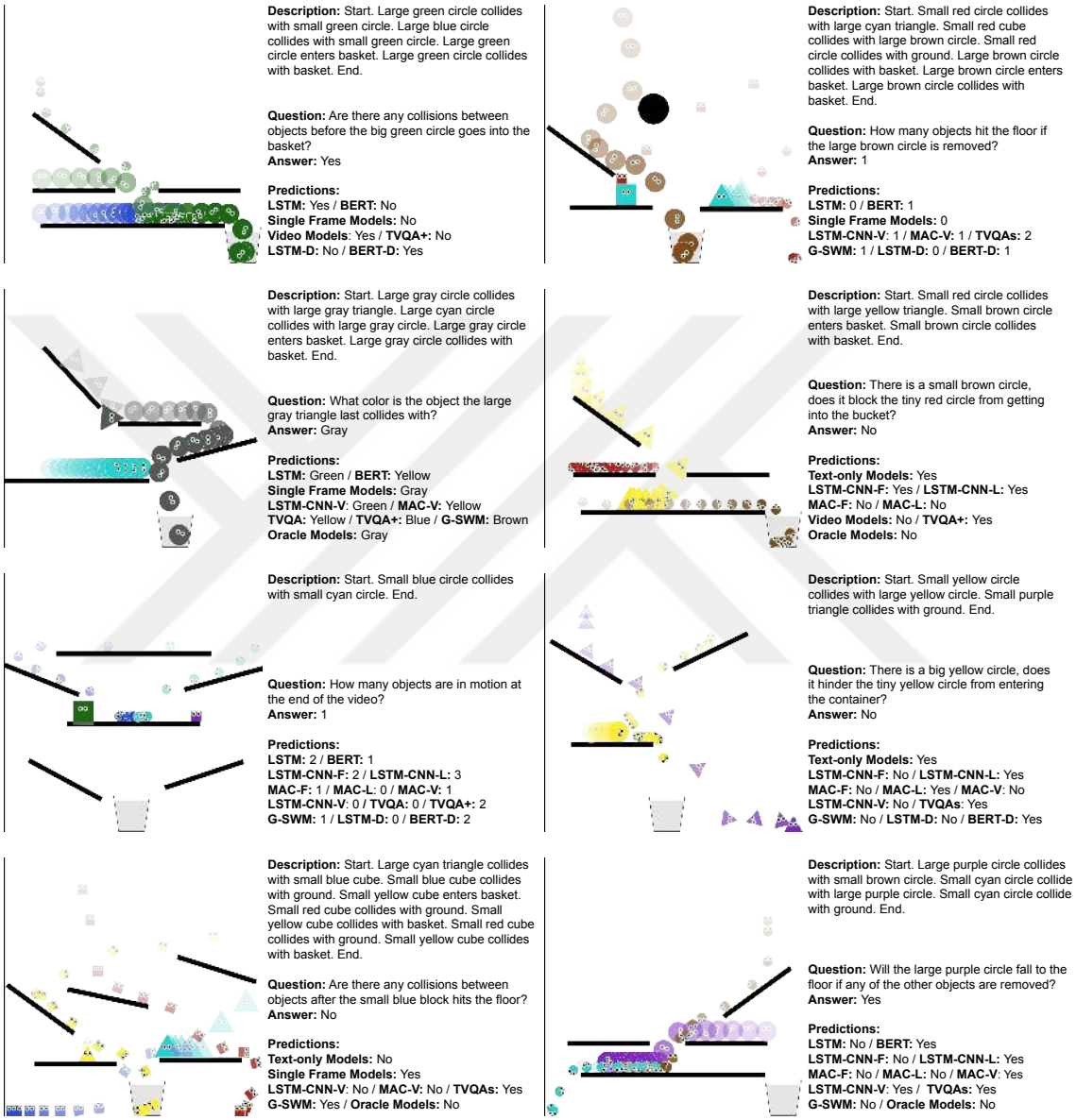


Figure 3.3: Selected baseline model predictions. The examples on the left belong to the descriptive category and the right column contains examples from the other categories.

3.5 Related Work

3.5.1 Visual Question Answering

Previously existing visual question answering (VQA) datasets can be categorized along two dimensions. The first dimension is the type of visual data, which includes either real world images [Malinowski and Fritz, 2014, Ren et al., 2015, Antol et al., 2015, Zhu et al., 2016b, Goyal et al., 2017b] or videos [Tapaswi et al., 2016, Lei et al., 2018], or synthetically created content [Johnson et al., 2017, Zhang et al., 2016a, Yi et al., 2020]. The second is at how the questions and answers are collected, which are usually done via crowdsourcing [Malinowski and Fritz, 2014, Antol et al., 2015] or by automatic means [Ren et al., 2015, Lin et al., 2014a, Johnson et al., 2017]. A key challenge for creating a good VQA dataset lies in minimizing the dataset bias. A model may exploit such biases and cheat the task by learning some shortcuts. In CRAFT, [Ates et al., 2022] generates questions about simulated scenes using a pre-defined set of templates by considering some heuristics to eliminate strong biases. Compared to the existing VQA datasets, CRAFT is specifically designed to test models' understanding of dynamic state changes of the objects in a scene. Although some prior work focuses on temporal reasoning [Lei et al., 2018, Yu et al., 2019, Lei et al., 2020b, Girdhar and Ramanan, 2020], they do not require the models to have a deep understanding of physics and/or imagine the consequence of certain actions to answer the questions, the only exceptions being TIWIQ [Wagner et al., 2018], CLEVRER [Yi et al., 2020], CLEVR_HYP [Sampat et al., 2021] and TVR [Hong et al., 2021] datasets. In these datasets, there exist hypothetical questions that require mental simulations about the consequences of performing certain actions or the lack of specific actions or objects. These datasets have received interest in developing neuro-symbolic reasoning models with physical understanding capabilities [Ding et al., 2020, Chen et al., 2021, Ding et al., 2021b]. CRAFT shares a similar design goal with the aforementioned datasets – but the scenes in the CRAFT benchmark are temporally more complex.

3.5.2 Understanding Physics in Artificial Intelligence

Lately, there has been a growing interest within the community in developing datasets and models to evaluate the ability of understanding and reasoning about the physical world. A notable amount of these efforts focuses on physical scene understanding. For instance, some researchers have explored the problem of predicting whether a set of objects are in stable configuration or not [Mottaghi et al., 2016] or if not where they fall [Lerer et al., 2016]. Others have tried to estimate a motion trajectory of a query object under different forces [Mottaghi et al., 2016] or developed methods to build a stack configuration of the objects from scratch through a planning algorithm [Janner et al., 2019]. [Li et al., 2019] suggested to represent rigid bodies, fluids, and deformable objects as a collection of particles and used this representation to learn how to manipulate them. Recently, [Bakhtin et al., 2019] and [Allen et al., 2020] created the PHYRE and the Tools benchmarks, respectively, which both include different types of 2D environments. An agent must reason about the scene and predict the outcomes of possible actions in order to solve the task associated with the environment. CoPhy [Baradel et al., 2020] is another recent work, which deals with physical reasoning prediction about counterfactual interventions. Although these works involve complicated physical reasoning tasks, the language component is largely missing. As mentioned, [Wagner et al., 2018], [Yi et al., 2020] and [Sampat et al., 2021] created VQA datasets for intuitive physics, but they lack visual variations unlike PHYRE and Tools. Though less studied, there are also some efforts in the NLP community to evaluate physical reasoning abilities of language models. [Bisk et al., 2020b] proposed the PIQA dataset that involves a binary choice task about daily activities regarding physical commonsense. Similarly, [Aroca-Ouellette et al., 2020] presented the PROST benchmark which includes questions that are designed to probe language models in a zero-shot setting and focuses on concepts like gravitational forces, physical attributes and object affordances.

The CRAFT dataset aims to combine the best of both worlds. In addition to the two types of questions investigated in CLEVRER [Yi et al., 2020], namely *descriptive* and *counterfactual*, CRAFT also includes questions that need reasoning about *causal*

interactions through the concepts like *cause*, *enable*, and *prevent*. To succeed in these tasks, models need to learn the semantics of each verb category that specifies different kinds of object interactions and their outcomes, i.e. to gain an understanding of a kind of commonsense knowledge.

3.6 Conclusion

In this chapter, we investigated how well the existing vision-language models can answer the questions about complex dynamic scenes. To do so, we chose the recent CRAFT benchmark as our test suite, and benchmark a variety of models. We showed that the existing models struggle to reason about physical forces and object interactions, and fall behind humans by a large margin. This points out a substantial room to improve these models. We also found that the existing models could be hallucinating by predicting colors which are absent in the scenes. As a limitation, we did not report the results of recent neuro-symbolic models, e.g. NS-DR [Yi et al., 2020]. Such approaches are very compelling and worth pursuing, but they currently require extra object-level annotations.

Most importantly, our oracle description baselines evaluated on the CRAFT resemble *chain-of-thought* (CoT) prompting approaches [Wei et al., 2022], a knowledge-augmentation methodology for complex problems where intermediate reasoning is mandatory. Our results suggest that the implemented video-language models cannot capture events taking place in videos, whereas an oracle baseline accessing this knowledge as a CoT prompt shows remarkable performance, reaching near-human accuracy. This finding implies that to bridge the gap between vision and language modalities in such a problem, the community should devise more proficient methodologies. In this direction, object-centric methods like G-SWM could be a viable solution.

Chapter 4

PROBING ACTION COUNTING AND RECOGNITION CAPABILITIES OF VIDEO-LANGUAGE MODELS

4.1 Introduction

In this chapter, we[1] propose a zero-shot evaluation benchmark for pretrained Video-Language Models (VidLMs) to better probe their language understanding abilities. VidLMs have received increasing attention from the research community [Lei et al., 2021, Luo et al., 2022, Xu et al., 2021a, Zellers et al., 2021b, Luo et al., 2020c, Fu et al., 2021, Ma et al., 2022, Bain et al., 2021, Ge et al., 2022, Lei et al., 2022, Zhu et al., 2022b, Cheng et al., 2023]. VidLMs can address questions previously unanswerable with image-language models (ILMs), since VidLMs can directly capture dynamically evolving phenomena such as events, natural/physical processes and actions, providing them with increased potential to perform commonsense reasoning [Zellers et al., 2021b]. This additional *temporal dimension* available to VidLMs introduces new challenges, since now models must not only keep track of objects and agents during points in time, but also account for changes caused by (or to) objects and agents.

VidLMs are commonly evaluated across different tasks such as video captioning [Yu et al., 2016a], text-to-video retrieval [Wang et al., 2021], and video question answering [Yu et al., 2019]. Such evaluations shed light on task performance and support comparative analysis. However, they are limited in their ability to reveal the specific visuo-linguistic capabilities models exhibit *across tasks*.

To bridge this gap, we propose a task-independent benchmark that focuses on two visio-linguistic phenomenons, namely *repetitive action counting* and *rare action*

[1]**Disclaimer:** This is a collaborative project with others. I only included the parts where I am the main contributor. This work is still in process, and could lack in some parts.

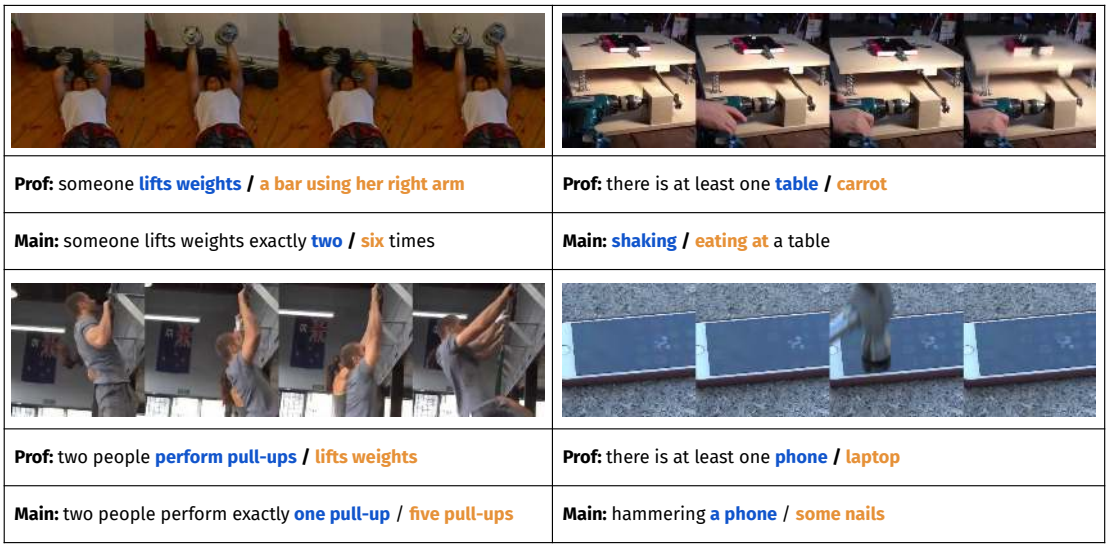


Figure 4.1: Some selected examples from the dataset, presented with their proficiency and main tests. Action counting examples are presented on the left, and rare action examples on the right. The former rare action example demonstrates action replacement, and the latter demonstrates object replacement.

recognition. We build upon the idea of VALSE [Parcalabescu et al., 2022] which is a similar benchmark for ILMs, with a focus on specific linguistic phenomena that are reflected in vision. As different from VALSE, our benchmark targets VidLMs, where *temporal reasoning* and *commonsense reasoning* are our primary focus. Similar to VALSE, we adopt a common structure for each *test*[2]: (i) We harvest high-quality examples from existing video-language datasets; (ii) we create counterfactual examples, i.e. foils, [Shekhar et al., 2017b], where only a small change is made to an existing example, so we can measure how well a model performs on the *test*, (iii) we apply automatic and manual validation of the examples and their foils to control for biases and to ensure a high-quality evaluation benchmark[3]; (iv) finally, we test whether existing VidLMs can distinguish correct from foils for given input videos.

[2]We choose *test* term over *piece* term used in VALSE to denote the benchmark tasks.

[3]This part is *work in progress*: We have been waiting for the human validation study to finish.

In addition to these, we also introduce proficiency tests in our evaluations. They test criteria that can be considered as preconditions for solving the main tests, by assessing the VidLMs' capability to successfully navigate and solve simpler objectives before attempting the more intricate and demanding main tests. We show that proficiency tests leads to significant performance decreases, suggesting that many apparently correct predictions by VidLMs can be accidental or spurious.

The rest of this chapter is structured as follows. Section 4.2 and Section 4.3 introduce the proposed tasks for this benchmark in detail, which are repetitive action counting and rare action recognition. In Section 4.4, we describe our experimental setup in detail and share the results of the conducted experiments. We briefly review the relevant literature in Section 4.5. Section 4.6 lists our contributions and limitations of this work.

4.2 Repetitive Action Counting

In this section, we share the details of the benchmark creation procedure of the counting task. In Table 4.1, we show a pair of examples. The **counting** task aims to probe the ability of models to accurately count the occurrences of *actions* within a given input video. Distinct from its image-based counterpart in the prior work VALSE [Parcalabescu et al., 2022] which examines the occurrences of *objects*, this task requires spatio-temporal reasoning, presenting a novel and interesting challenge. Section 4.2.1 contains the details of our data sources. In Section 4.2.2, we document our foil creation strategy, together with subsets presented for this task.

4.2.1 Data Sources

We use the QUVA dataset [Runia et al., 2018], comprising 100 videos. Within each video, every occurrence of the target action is annotated with a corresponding frame number, specifying the end of each action. The QUVA dataset lacks any textual annotations. Consequently, we curate multiple textual templates per video, incorporating a placeholder for the numerical value (*¡number¿*). Emulating the approach in VALSE, our templates incorporate the term *exactly* to indicate precise

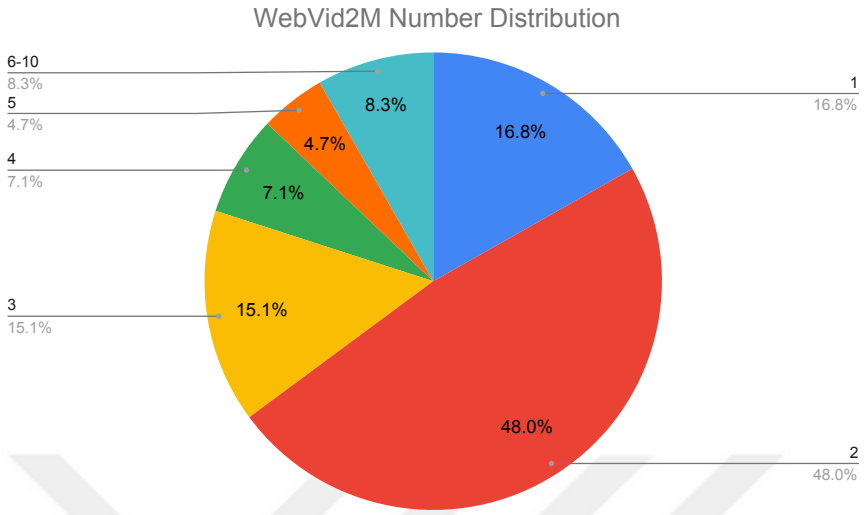


Figure 4.2: WebVid2M dataset [Bain et al., 2021] number distribution. The indefinite articles (a/an) are opted out. Numbers 6, 7, 8, 9 and 10 are merged into single category *6-10*.

counting (e.g., someone performs exactly ¡number¿ push-ups). We take care to avoid overly specific terms, opting for more general descriptors (e.g., *lifting weights* instead of *skull-crushers arm exercise*). A native English speaker checked the manually collected templates and fixed potential syntax errors in them. We set the videos' frame per second rate to 30, since VideoCLIP [Xu et al., 2021a] only works with 30-FPS videos. We choose to use the spelled-out numbers (e.g. one, two etc.) rather than numerical numbers (e.g. 1, 2 etc.).

4.2.2 Foiling Method

To create captions and foils, we replace the number placeholder with the correct numerical value and an incorrect one. We discard all instances with counts exceeding a predetermined threshold T_c, set at 10. For the counting task, we created two subtasks: the **easy** and the **difficult**. In the **easy** subtask, we deliberately opt for small numbers $C \in \{1, 2, 3\}$ in the captions. The choice of these small numbers aligns with the notion that models frequently encounter such quantities during pretraining

(see Figure 4.2), making them more recognizable and interpretable. By contrast in the **difficult** sub-tests, we favor small numbers in the foils. This presents a challenging task for VidLMs as it tests the models' ability to overcome any bias towards numbers frequently encountered during pretraining. In this way, we aim to assess the models' true abilities to handle counting in diverse contexts.

4.2.3 Proficiency Tests

The counting task's proficiency tests assess how well the models recognize the actions repeated in the videos. To create the proficiency captions, we remove number-specific phrases. For instance, we change "a man performs exactly ¡number¿ push-ups." to "a man performs push-ups.". To create proficiency foils, we implement a procedure that has 4 main stages which are,

1. We use spaCy's[4] dependency parser to localize the verb phrases. We mask these phrases and generate text for the masked spans using T5 (t5-large) [Raffel et al., 2020]. To obtain the initial foil candidates, we filter out generations that include personal pronouns (e.g. I, they etc.) and conjunctions (e.g. and, but). We then perform GRUEN and NLI filtering [Zhu and Bhat, 2020]. Similar to the other tasks' proficiency task, we discard candidates that have a GRUEN score lower than a certain threshold which is equal to 0.80. We employ NLI filtering and filter out the examples that *entails* the proficiency caption. As NLI model, we use a fine-tuned Albert (`ynie/albert-xxlarge-v2-snli_mnli_fever_anli_R1_R2_R3-nli`) in Huggingface's transformers package [Lan et al., 2020, Wolf et al., 2019]. As a last step, we perform manual intervention and discard implausible foil candidates.

2. We mask the subject and noun phrases in the captions and then we repeat the first step using RoBERTa [Liu et al., 2019b] for the examples that do not have a single foil. We repeat the first stage's GRUEN/NLI and manual filtering steps again.

[4]spacy.io/

3. For the videos without any valid foils, we randomly sample captions from the other videos. We restrict this sampling process categorically: For the exercise videos, we only sample captions from other exercise videos. We exclude the captions that comprise "*the same exercise*" phrase. We then replace the subject phrases with the ground-truth caption's subject phrase to make it similar to the true caption. We perform an NLI filtering as a final step to finalize the foil candidates.

4. To obtain the foil, we randomly sample from the candidate set.

The reason why we employ this 4-stage procedure is the captions' degree of specificity for some examples. For instance, if we mask the verb or noun phrases of the sentence "*a man performs push-ups.*", LMs naturally fail to come up with different phrases. We can observe the same phenomena for sentences "a kid jumps on a trampoline." and "somebody pushes a button".

4.3 Rare Action Recognition

In the **rare actions** tests, we investigate the ability of VidLMs to identify novel compositions and recognize unusual events, such as a computer keyboard is being cut using a chainsaw by someone described. Figure 4.1 shares a pair of examples. These events are described by a verb-noun pair, e.g. "*cutting a keyboard*". We choose foils from more likely events taking place in videos. Section 4.3.1 contains the details of our data sources. In Section 4.3.2, we document our foil creation strategy, together with subsets presented for this task. Section 4.2.3 describes how we create the proficiency tests for the rare actions tests.

4.3.1 Data Sources

We leverage the RareAct dataset [Miech et al., 2020], which consists of videos accompanied by action-object pairs describing events within the videos. These action-object pairs are extracted by analyzing co-occurrence statistics from the

widely used HowTo100M [Miech et al., 2019] dataset for VidLM pretraining. The RareAct dataset does not have a natural language component: It only tests action recognition models. To enrich this dataset, we generate simple captions based on the action-object pairs. For instance, given the action-object pair *cut-keyboard*, we create the descriptive caption *cutting a keyboard*. We avoid subject phrases in captions since some videos do not comprise a human being as an actor.

4.3.2 Foiling Method

This task offers two sub-tests: the **action replacement** and the **object replacement** sub-test. In the **action replacement** sub-test, we substitute the original action with a more plausible alternative that can be applied to the given object, e.g. *type on* for the previous *keyboard* example. To generate foils in this sub-test, we employ T5 [Raffel et al., 2020], as it enables us to produce foil candidates with *compound verbs*, e.g., *talk on*, *place at*, etc. We discard foil candidates with some general actions (e.g. use, have etc.) or actions that imply some form of *touching* (e.g. hold, reach etc.). We then perform an NLI and GRUEN score filtering. In this stage, we perform a manual intervention and abandon the low quality candidates. As for the **object replacement** sub-test, we focus on replacing the object in the action-object pair. For instance, revisiting the previous example, we replace the object *keyboard* with *bread*. Here, we prefer to use a set of token-based MLMs [Devlin et al., 2019, Lan et al., 2020, Liu et al., 2019b]. To further enhance the quality of the foils, we opt for an ensembling approach in the object replacement sub-test. Particularly, we use three MLMs which are BERT, RoBERTa and ALBERT [Devlin et al., 2019, Liu et al., 2019b, Lan et al., 2020]. We also run an object detector called End-to-End Object Detection model (DETR) [Carion et al., 2020] and discard the foil candidates that contain the detected objects. We use `facebook/detr-resnet-101` DETR in Huggingface's transformers package [Wolf et al., 2019]. We uniformly sample $K = 8$ frames and run the object detector on these sampled frames. We classify an object detected if its confidence threshold exceeds 0.80.

4.3.3 Proficiency Tests

The proficiency test of the rare actions tests the existence of objects in the given input videos, similar to the VALSE existence instrument. This time we do not use negated foils: We replace the correct object with another. To create captions, we create a statement about the existence of the ground truth object, e.g. "*there is at least one keyboard.*". To create foils, we randomly sample from the objects appear in ground-truth captions, e.g. "*there are some flowers*". Similar to the main test, we implement the same object detection filtering process.

4.4 Experiments

This section describes our experimental setup, results and analyses. §4.4.1 provides an overview of the models tested on our benchmark. §4.4.5 contains the implementation details of these models. §4.4.6 describes the evaluation metrics used to measure model performances. §4.4.7 presents the obtained results and our key observations.

4.4.1 Pretrained Models

Here we describe the models used in this benchmark. Next to the pretrained video-language models (§4.4.4), we also experimented with pretrained unimodal models (i.e. text-only LMs) and image-language models (§4.4.2 and §4.4.3).

4.4.2 Unimodal Models

We test a couple of decoder-only or encoder-decoder LMs on the benchmark. These models are GPT-2 [Radford et al., 2019], OPT [Zhang et al., 2022], T5 [Raffel et al., 2020] and BART [Lewis et al., 2020]. Similar to VALSE, we calculate the perplexity values for both caption and foil, and select the text input with smaller perplexity score. During our experiments for GPT-2 and OPT, gpt2[5] with 124M parameters and opt-6.7b[6] are used.

[5] `https://huggingface.co/gpt2`

[6] `https://huggingface.co/facebook/opt-6.7b`

4.4.3 Image-Language Models

We also conducted experiments involving two prominent Image-Language models CLIP [Radford et al., 2021] and BLIP-2 [Li et al., 2023c]. CLIP employs a dual-encoder architecture, utilizing a contrastive loss objective to facilitate the training of image-caption pairs. On the other hand, BLIP-2 represents a subsequent advancement of BLIP [Li et al., 2022b], harnessing the potential of frozen pretrained image encoders and large language models to bolster the vision-language learning process. For CLIP and BLIP-2 experiments, the largest version of CLIP[7] and BLIP-2[8] with OPT-6.7B version are used.

4.4.4 Video-Language Models

In this section, we share the details of the pretrained video-language models used for the experiments. §4.4.5 shares the implementation details of these models.

VideoCLIP [Xu et al., 2021a] uses BERT as text encoder and S3D [Xie et al., 2018] as video encoder. VideoCLIP is pretrained on HowTo100M. Like ClipBERT, it uses mean pooling to fuse modalities.

FiT [Bain et al., 2021] encodes text using BERT like many others. As video encoder, TimeSFormer [Bertasius et al., 2021] is preferred. FiT is pretrained on both images (CC3M) and videos (W2). It creates a shared video-text space via contrastive learning. The authors also collected the W2 dataset.

X-CLIP [Ma et al., 2022] is a video-text retrieval model that offers a new approach to address the challenge of similarity aggregation. By employing a multi-grained contrastive mechanism, the model encodes sentences and videos into coarse-grained and fine-grained representations, facilitating contrasts across different levels of granularity. Moreover, the model introduces the Attention Over Similarity Matrix (AOSM) module, enabling it to focus on essential frames and words while reducing the impact of irrelevant ones during retrieval.

---

[7] `https://huggingface.co/openai/clip-vit-large-patch14`

[8] `https://huggingface.co/Salesforce/blip2-opt-6.7b`

MCQ [Ge et al., 2022] introduced a pretext task as Multiple Choice Questions (MCQ) for video-text pre-training based on a dual-encoder mechanism. They used a parametric module called BridgeFormer, which connects local features from VideoFormer [Dosovitskiy et al., 2020] and TextFormer [Sanh et al., 2019] to answer multiple-choice questions via contrastive learning objective. It enhances semantic associations between video-text representations and improves fine-grained semantic associations between two modalities. Additionally, it maintains high efficiency for retrieval and the BridgeFormer can be removed for downstream tasks.

Singularity [Lei et al., 2022] showed the effectiveness of single-frame training in the context of video-language tasks, such as video question answering and text-to-video retrieval, by incorporating a vision encoder [Dosovitskiy et al., 2020], a language encoder [Devlin et al., 2019], and a multi-modal encoder with cross-attention fusion mechanism. On the other hand, they have implemented a new benchmark to overcome focusing on models temporal learning abilities. This contribution brings to light a significant static appearance bias prevalent in current video-and-language datasets.

VindLU [Cheng et al., 2023] followed a comprehensive approach for enhancing VidLM pretraining to fine the most effective VidLM framework design. The methodology begins by employing image [Bao et al., 2021] and text [Devlin et al., 2019] encoders, trained on video and caption pairs through a visual-text contrastive objective. Subsequently, the framework progressively incorporates additional components while analyzing the significance of each one. The final recipe encompasses six steps, which involve the inclusion of temporal attention, integration of a multi-modal fusion encoder, adoption of masked modeling pretraining objectives, joint training on images and videos, utilization of additional frames both in fine-tuning and inference stages, model-parameter and data scaling. These steps collectively contribute to an effective VidLM pre-training process, facilitating improved performance and understanding in multi-modal video question answering tasks.

4.4.5 Implementation Details

We try to use each model *as-is* based on the provided official implementations in a zero-shot setting. We directly use Huggingface implementations [Wolf et al., 2019] of GPT-2, OPT, CLIP, BLIP2 and X-CLIP. The majority of VidLMs sample a model-specific number of frames K to construct video input. Specifically, X-CLIP use $K = 8$ whereas the remaining tested models use $K = 4$. VideoCLIP processes the entire video using a S3D video encoder [Xie et al., 2018]. To calculate video-caption match scores for ILMs, we perform mean pooling over the image-caption match scores obtained using multiple frames, setting $K = 8$ following the X-CLIP implementation. We run experiments on single Tesla T4 or V100 GPUs using half precision.

4.4.6 Evaluation Metrics

Following [Parcalabescu et al., 2022], we use the **pairwise ranking accuracy** acc_r metric to compare models, since only this metric allows us to evaluate VidLMs pretrained with the VTC, VTM and NLG objectives at the same time. We report scores for the main tests (T) and their respective proficiency tasks (P). We also report a combined score (C), whereby a model which succeeds on the main test is only considered correct if it also succeeds on the relevant proficiency test.

4.4.7 Results and Discussion

Counting Results

Table 4.1 presents the model performances achieved on the counting tests. All of the adapted models perform close to a random baseline in the overall main tests. In the proficiency tests, ILMs and VidLMs achieve a decent performance. However, we observe a significant decrease in performance when evaluating the models under combined settings, ranging from 12.50% to 50% proportional to their main test performances.

We also conducted a categorical evaluation, where each category contains examples that have specified count in their ground truth captions. Figure 4.3 shows these

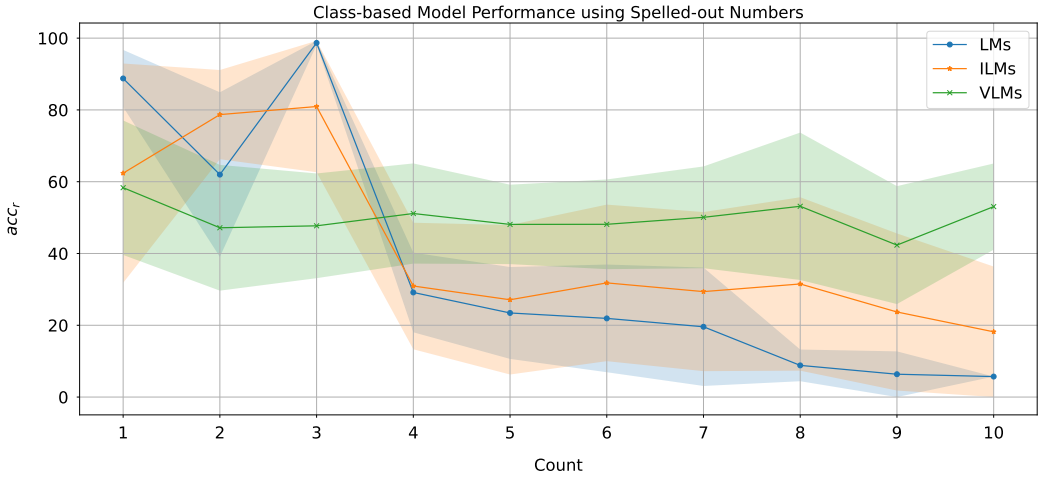


Figure 4.3: Categorical evaluation on the counting main task.

class-based analysis. To simplify this analysis, we compute average performances for each model category, unimodal LMs, ILMs and VidLMs. The standard deviation values are illustrated with the color filled areas. We observe that LMs and ILMs are heavily biased towards smaller numbers. This is as expected for the unimodal LMs, since they don't process visual input. Most importantly, the implemented VidLMs perform close to a random baseline, indicating that these models are unable to count actions.

Rare Actions Results

Table 4.2 presents the model performances achieved on the rare actions tests. Unimodal baselines GPT2 and OPT [Radford et al., 2019, Zhang et al., 2022] perform very poorly on the main tests as expected since the captions describe less likely events. We observe that the models consistently perform better in object replacement tests in comparison to action replacement. This is inline with previous work where the models are more biased towards nouns and they fail to process verbs sufficiently [Momeni et al., 2023, Park et al., 2022, Lei et al., 2022]. Additionally, CLIP outperforms all

Model	**Easy**			**Difficult**			**All**		
	P	T	C	P	T	C	P	T	C
Random	50.00	50.00	25.00	50.00	50.00	25.00	50.00	50.00	25.00
GPT2	48.85	71.27	34.53	49.25	33.55	18.06	49.04	53.08	26.59
OPT	56.86	93.89	53.05	57.42	10.75	6.77	57.13	53.55	30.74
CLIP	89.89	65.67	45.95	90.11	38.60	46.13	90.00	52.62	46.04
BLIP2	81.38	93.99	75.88	83.12	9.03	7.85	82.22	53.03	43.08
VideoCLIP	78.58	31.33	25.23	80.65	60.75	48.28	79.58	45.51	36.34
FiT	83.98	52.95	44.74	82.90	49.57	41.94	83.46	51.32	43.39
MCQ	82.32	29.03	26.33	82.04	71.94	56.77	82.19	49.72	41.01
X-CLIP	85.99	65.87	55.26	84.52	41.72	36.02	85.28	54.23	45.98
Singularity	80.78	56.16	45.35	80.43	46.45	38.06	80.61	51.48	41.84
Singularity-T	79.78	60.26	47.35	79.14	41.72	34.73	79.47	51.32	41.27
VindLU	85.19	66.57	58.16	83.33	35.27	28.17	84.29	51.48	43.70

Table 4.1: Performance of the adapted models on the counting task using pairwise ranking accuracy (acc_r) metric on the **P**roficiency, main **T**ask and **C**ombined tests. Each row represents a different model. Easy, Difficult and All stands for the results achieved on the corresponding splits.

VidLMs except VindLU, which demonstrates its ability to handle novel compositions is superior to most VidLMs. Similar to the counting tests, we observe a significant decrease when we include proficiency tests in evaluation. In the combined results, most evaluated models fail in 20% of the examples in this task.

Model	**Action Repl.**			**Object Repl.**			**All**		
	P	T	C	P	T	C	P	T	C
GPT2	55.20	17.50	8.50	54.80	34.50	25.60	55.00	26.00	17.00
OPT	58.00	20.30	11.30	59.90	27.40	20.30	58.95	23.85	15.80
CLIP	92.30	93.60	86.50	92.90	93.60	89.50	92.60	93.60	88.00
BLIP2	90.40	64.20	58.80	92.90	83.90	80.00	91.65	74.05	69.00
VideoCLIP	81.50	75.50	61.30	84.00	81.10	72.00	82.75	78.30	66.65
FiT	87.80	86.60	75.40	87.80	91.20	82.90	87.80	88.90	79.15
MCQ	89.60	86.30	76.70	89.50	90.00	83.60	89.55	88.15	80.15
X-CLIP	83.10	86.00	70.70	85.20	85.00	75.30	84.15	85.50	73.00
Singularity	92.20	86.20	79.20	93.00	90.70	86.10	92.60	88.45	82.65
Singularity-T	90.30	85.90	76.80	92.40	92.00	87.10	91.35	88.95	81.95
VindLU	93.70	92.30	86.20	94.40	93.70	89.90	94.05	93.00	88.05

Table 4.2: Performance of the adapted models on the rare actions task using pairwise ranking accuracy (acc_r) metric on the **P**roficiency, main **T**ask and **C**ombined tests. Each row represents a different model.

4.5 Related Work

This section summarizes related work by categorizing pretrained video-language models (VidLMs) (§4.5.1), reviewing recent efforts that investigate the capabilities of pretrained image-language models (ILMs) (§4.5.2) and positioning our work in relation to existing similar video-language benchmarks (§4.5.3).

4.5.1 Pretrained VidLMs

We categorize VidLMs along distinct aspects: pretraining of the visual modality, the pretraining datasets, pretraining objectives, temporal modeling and multi-modal fusion strategies. See §4.4 for detailed descriptions of models used in our experiments.

Modalities. Pretraining of the visual modality can be performed on images [Lei et al., 2021], videos [Li et al., 2020, Zhu and Yang, 2020, Xu et al., 2021a, Zellers et al., 2021b, Seo et al., 2022, Wang et al., 2022a, Li et al., 2022a, Luo et al., 2022] or both [Bain et al., 2021, Fu et al., 2021, Wang et al., 2022b, Li et al., 2022c, Lei et al., 2022]. A handful of models [Akbari et al., 2021, Lin et al., 2022, Zellers et al., 2022] also incorporate the auditory modality, i.e. sound.

Datasets. The chosen pretraining datasets often depend on the type of pretraining done in the visual modality. Early VidLMs (e.g. [Zhu and Yang, 2020, Li et al., 2020, Xu et al., 2021a]) use HowTo100M [Miech et al., 2019], which provides the linguistic modality in the form of Automatic Speech Recognition (ASR) output or manually written subtitles. Recent models are pretrained on the WebVid-2M (W2) dataset [Bain et al., 2021], which follows a similar approach to CC3M [Sharma et al., 2018] in filtering items based on the quality of the textual modality. In addition to video-text datasets, recent VidLMs leverage also large-scale image-text datasets such as SBU captions [Ordonez et al., 2011], Conceptual Captions (CC) 3M and CC12M [Changpinyo et al., 2021].

Objectives. Some pretraining objectives for VidLMs have been derived from the pretraining objectives employed by ILMs. The most prominent among these are video-text contrastive loss (VTC), video-text matching (VTM), masked language modeling (MLM) and masked frame modeling (MFM). A few exceptions employ natural language generation (NLG) [Seo et al., 2022, Wang et al., 2022b], masked visual-token modeling (MVM) [Li et al., 2022c] or temporal reordering [Zellers et al., 2021b].

Temporal Modeling. Only a few methods use joint space-time attention [Bertasius et al., 2021, Bain et al., 2021, Wang et al., 2022b] to process video. Some approaches [Zellers et al., 2021b, Luo et al., 2022, Yang et al., 2022] rely on language at this

stage, and implement a multi-modal attention mechanism between patches and word embeddings. [Fu et al., 2021, Li et al., 2022c] extract spatio-temporal features using the Video Swin Transformer [Liu et al., 2022] with shifted window attention [Liu et al., 2021b].

Multi-modal Fusion. Models relying exclusively on the VTC objective disregard multi-modal fusion [Xu et al., 2021a, Bain et al., 2021, Luo et al., 2022, Lin et al., 2022]. Others either implement an additional multi-modal transformer [Luo et al., 2020c, Lei et al., 2022, Seo et al., 2022] or create a visual prefix to be fused into a text-only LM [Zellers et al., 2021b, Fu et al., 2021].

4.5.2 Benchmarks for Pretrained ILMs

Image-Language Models (ILMs) are commonly tested on *tasks* such as image question answering [Goyal et al., 2017b], visual reasoning [Suhr et al., 2019] or image retrieval [Lin et al., 2014a, Plummer et al., 2015]. Some benchmarks measure *task-overarching capabilities* of ILMs, such as their understanding of actions [Hendricks and Nematzadeh, 2021] or compositionality [Thrush et al., 2022]. A specific way of testing ILMs is *foiling* [Shekhar et al., 2017b, Gokhale et al., 2020, Bitton et al., 2021, Parcalabescu et al., 2021, Rosenberg et al., 2021], where a caption is turned into a counterfactual (i.e., *foil*) by minimal edits, such that it does not correctly describe the image anymore [Shekhar et al., 2017b, Shekhar et al., 2017a]. As an alternative, the image can be exchanged, such that it does not match the caption anymore [Rosenberg et al., 2021, Wang et al., 2023b]. The key consideration in creating counterfactuals is to target the linguistic elements in focus, which are assumed to reflect specific model capabilities (e.g. by altering a preposition, a model's ability to distinguish caption from foil should reflect its understanding of spatial relations). An alternative strategy is to test a large pretrained model on multiple choice questions designed to probe specific capabilities, as recently used in SEED-Bench [Li et al., 2023a].

Most related to our work is the VALSE foiling benchmark [Parcalabescu et al., 2022], which tests the linguistic grounding capabilities of ILMs. It targets specific

linguistic phenomena: existence, plurality, counting, spatial relations, actions, and entity coreference. The models are tested in zero-shot mode on the image-text alignment task, which is one of the objectives in ILM pretraining. [Bugliarello et al., 2023] tested current encoder-only ILMs on the benchmarks mentioned above, namely SVO probes [Hendricks and Nematzadeh, 2021], VALSE [Parcalabescu et al., 2022], and Winoground [Thrush et al., 2022].

4.5.3 Benchmarks for Pretrained VidLMs

Like ILMs, VidLMs are evaluated on numerous downstream tasks, primarily action recognition [Kuehne et al., 2011, Soomro et al., 2012], video-text retrieval [Xu et al., 2016, Hendricks et al., 2017], and video question answering (VidQA) [Xu et al., 2017, Lei et al., 2018]. [Lei et al., 2022] shows that a non-temporal model can perform better than temporal models in these benchmarks. Newer VidQA benchmarks [Lei et al., 2020a, Xiao et al., 2021] offer stronger tests for VidLMs to probe their temporal reasoning and commonsense reasoning capabilities. In our benchmark, we also prioritize these aspects, but we cast the tasks in a zero-shot setting using a counterfactual setup, to elicit the pretrained models' inherent capabilities.

Foiling benchmarks have also been proposed to evaluate VidLMs. [Park et al., 2022] devise two tests. In the first one, the foils are created by swapping the character entities in the caption. In the second, an LM replaces the verb phrase of the caption. On the other hand, [Bagad et al., 2023] create a benchmark consisting of synthetic video-caption-foil triplets (e.g. a red circle appears after/before a yellow circle) to test how well VidLMs localize the events happening in the video. [Bagad et al., 2023] also propose a *consistency* test to probe whether the models localise the events correctly or just predict the correct answers. In St.Viola, we have a test similar to [Park et al., 2022], but we built it upon the semantic role labelling (SRL) task. Similar to the consistency task of [Bagad et al., 2023], we propose a *proficiency* task for each task. In contrast to these earlier foiling benchmarks, our benchmark St.Viola is more comprehensive as it is designed to examine models' grounding capabilities for different linguistic phenomena, by extending VALSE to the temporal axis.

Another notable benchmark is VALUE [Li et al., 2021b], which is developed in a fashion similar to the GLUE/SuperGLUE evaluation suites [Wang et al., 2019a, Wang et al., 2019b] proposed for natural language understanding. VALUE comprises 11 separate datasets covering 3 different downstream tasks. Unlike VALUE, our benchmark is a zero-shot *foiling* benchmark with particular focus on linguistic phenomena that emphasize temporal reasoning.

4.6 Conclusion

In this chapter, we have introduced a novel video-language foiling benchmark which offers two tasks: Repetitive Action Counting and Rare Action Recognition. This benchmark is designed to probe the capabilities of pretrained VidLMs, where we prioritized commonsense and temporal reasoning. We have conducted a comprehensive evaluation and comparison of numerous VidLMs as well as ILMs and text-only LMs on our benchmark. Our experiments show that, as far as visually grounded temporal reasoning abilities are concerned, VidLMs do not differ substantially from ILMs in action counting task. Our findings on the rare actions task demonstrated that VidLMs fall behind ILMs within atypical contexts. To further refine our benchmark, we have introduced proficiency tests, which not only enhance granularity but also provide deeper insights into the models' aptitude. Strikingly, our proficiency task results reveal that a considerable portion of correct predictions appears to be accidental rather than indicative of robust understanding. This highlights a critical point: current VidLMs might be struggling with the intricacies of temporal reasoning challenges offered by our benchmark. This further emphasizes the critical importance of benchmarks like ours, which serve to identify current weaknesses that need to be addressed in the drive towards more proficient VidLMs.

Chapter 5

EUPHEMISM DETECTION

5.1 Introduction

In this chapter, we[1] focus on a figurative language understanding task called *euphemism detection*. Euphemisms attempt to smooth harsh, impolite, or blunt expressions about taboo or sensitive topics like death and unemployment [Holder, 2008]. For instance, when we speak of older people we often refer to *senior citizens* instead of a direct expression that can be seen as offensive.

Identifying euphemisms is challenging due to their natural ambiguity, i.e., the meaning of the term shifts depending on the context: '*Over the hill*' could either mean someone or something is *physically* over some hill (*literal*), or someone or something is *old*, past one's prime (*figurative*) [Lee et al., 2022]. One cannot distinguish these two different senses without sufficient context. Thus, these terms are referred as *potentially euphemistic terms* (PETs) [Gavidia et al., 2022]. Here, we propose a two-stage method to detect euphemistic terms.

In the first stage, we manually collect literal descriptions for each PET. We then incorporate these descriptions into input text prompts to help the model distinguish figurative from literal usage. We practice this approach because PETs rarely appear together with their literal descriptions. We demonstrate that this kind of extraneous linguistic supervision improves a strong baseline by a large margin.

In the second stage, we supply extra visual supervision in addition to the extra linguistic supervision. We pursue this direction because extra modality could benefit language understanding in low-resource settings [Li et al., 2022d, Lu et al., 2022].

[1]I am the only main contributor in this study. This chapter is adapted from [Kesen et al., 2022b]

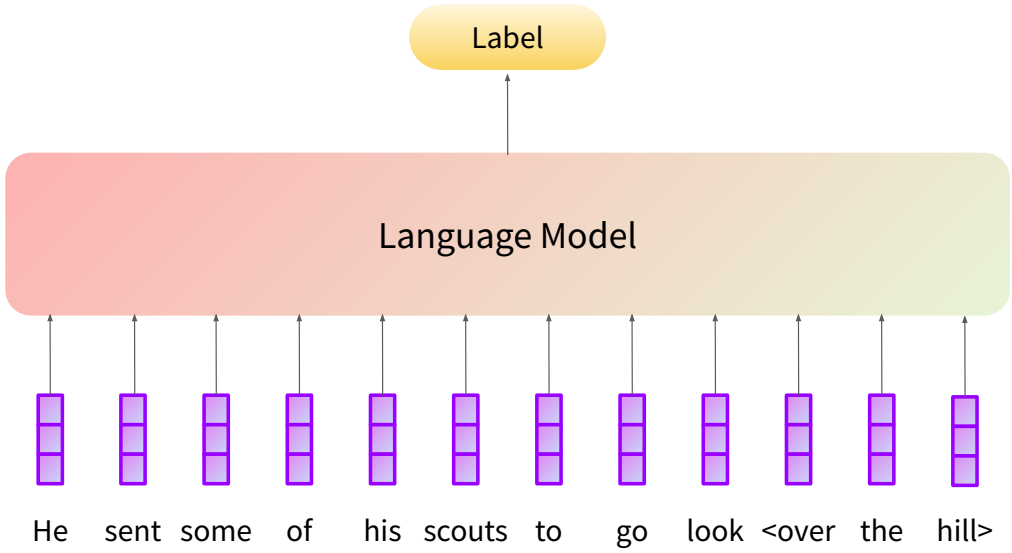


Figure 5.1: Visualization of our vanilla baseline approach.

We use a text-to-image model which takes terms and descriptions as input, and we generate two sets of images, which we denote as *visual imageries*. Our experiments show that using visual imagery provides the best results. A paired t-test points out that the improvement is statistically significant. Our qualitative analysis also suggests visual imageries are beneficial for analyzing PETs.

The rest of this chapter is organized as follows. Section 5.2 describes our proposed solution. In Section 5.3, we share the details of our evaluation setup and design choices. Section 5.4 reports our experimental results. In Section 5.5, we briefly review the relevant literature. Section 5.6 outlines our conclusions and discusses the limitations of our approach.

5.2 Approach

In this section, we first formulate the euphemism detection task by describing a simple baseline model, and then explain how we extend it with the literal term descriptions and visual imagery.

5.2.1 Vanilla Baseline

Given a textual context C with a potentially euphemistic term (PET) T, the aim of euphemism detection is to decide whether the candidate term T is euphemistic ($y = 1$) or not ($y = 0$). Here, we only pick a sentence $S = [w_1, w_2, ..., w_n]$ which contains a candidate term T, and ignore the rest of the context C at first. Figure5.1 illustrates our vanilla baseline. We use a pretrained language model LM as our initial baseline as described below.

$$e_i = \text{EMBED}(w_i)$$

$$\hat{p} = \text{LM}(e_1, e_2, ..., e_n)$$

$$\hat{y} = \begin{cases} 1 & \hat{p} \geq 0.5, \\ 0 & \text{otherwise.} \end{cases}$$

e_i denotes the word embedding of the i$^{\text{th}}$ token w_i, $\hat{p}$ is the probability that the candidate term T is euphemistic, and $\hat{y}$ is the predicted label. EMBED is the embedding layer and LM denotes the language model that produces the probability $\hat{p}$.

5.2.2 Literal Descriptions

We extend the baseline model by supplying extra supervision with literal descriptions D for each candidate term T (which we collect manually). Figure 5.2 illustrates how we incorporate these literal descriptions. To make use of the literal descriptions, we create a textual prompt $X = [x_1, x_2, ..., x_n]$ for each sentence S, term T and description D as below.

$$X = [\text{Term: } T, \quad \text{Description: } D, \quad \text{Sentence: } S].$$

Then, we change the formulation,

$$e_i = \text{EMBED}(x_i)$$

$$\hat{p} = \text{LM}(e_1, e_2, ..., e_n),$$

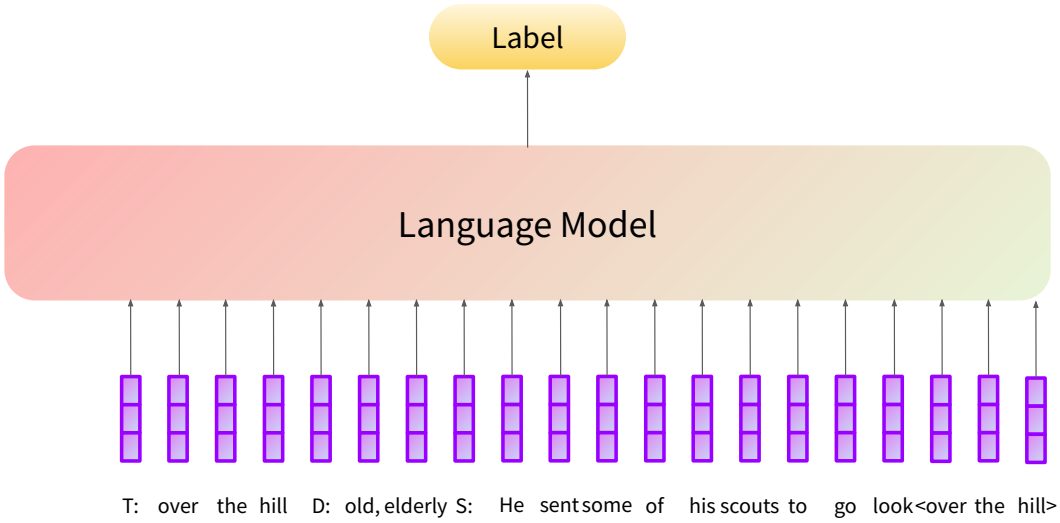


Figure 5.2: Illustration of our literal descriptions baseline. The literal descriptions baseline augments the language input with the literal descriptions of PETs.

where e_i is the embedding for the i$^{\text{th}}$ token of the input prompt X.

5.2.3 Visual Imagery

We subsequently move beyond the text-only baselines by integrating visual modality into the Literal Descriptions baseline in the form of *visual imagery*. To accomplish this, we generate two sets of images $I_T = [I_T^{(1)}, I_T^{(2)}, ..., I_T^{(k)}]$ and $I_D = [I_D^{(1)}, I_D^{(2)}, ..., I_D^{(k)}]$, for each term and description pair, respectively. We denote these set of images as visual imageries. To obtain the visual imageries, we feed a text-to-image model T2I with terms and descriptions as input language,

$$I_T^{(k)} \sim \text{T2I}(T), \qquad I_D^{(k)} \sim \text{T2I}(D).$$

Next, we use a pretrained visual encoder (VE) to embed visual imageries.

$$v_T = \frac{1}{K}\sum_{k=1}^{K} \text{VE}(I_T^{(k)}), \qquad v_D = \frac{1}{K}\sum_{k=1}^{K} \text{VE}(I_D^{(k)})$$

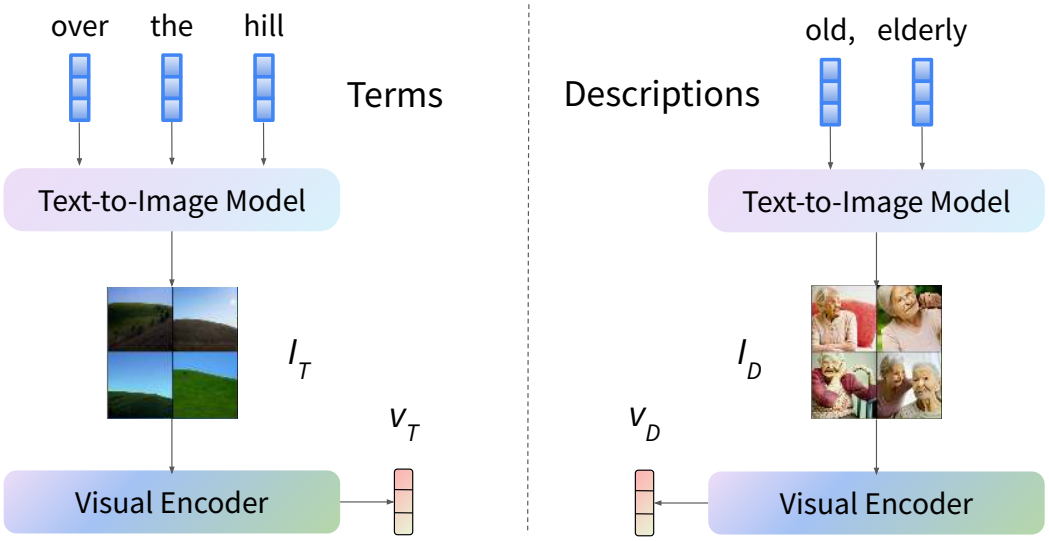


Figure 5.3: Illustration of our literal descriptions baseline. The literal descriptions baseline augments the language input with the literal descriptions of PETs.

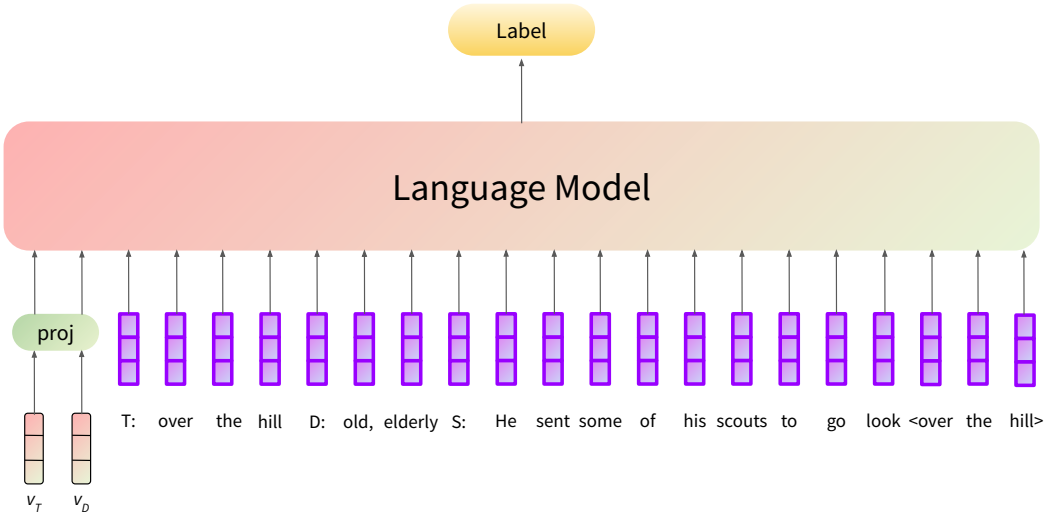


Figure 5.4: Illustration of our literal descriptions baseline. The literal descriptions baseline augments the language input with the literal descriptions of PETs.

where v_T denotes the visual imagery embedding of the candidate term T and v_D denotes the visual imagery embedding of the corresponding literal description D. K is the number of images per term T and description D. Figure 5.3 shows an overview of the visual imagery creation process. To make use of these imageries, we reformulate the literal descriptions baseline as follows,

$$e_i = \text{EMBED}(x_i)$$
$$\hat{p} = \text{LM}(f_p(v_T), f_p(v_D), e_1, e_2, ..., e_n)$$

We make sure visual imagery embeddings are compatible with the word embeddings and language model LM by applying a linear projection layer f_p. We train each baseline using the negative log-likelihood objective.

5.3 Data and Implementation

This section shares the details of the data and the implementation of our system.

5.3.1 Data

The euphemism detection dataset [Gavidia et al., 2022] consists of two separate splits for training and testing purposes with 1573 and 394 examples, respectively. The test split is unlabeled. The whole data includes 131 different PETs. Since there is no data supplied for validation, we reserve 20% of the training data for this purpose. We only select the sentences with PETs and remove repetitive patterns of punctuation *"@ @ @ ..."* to decrease computational requirements by shortening the input language. We manually collect literal descriptions within 6 hours, and try to avoid impolite expressions like insults or slang phrases.

5.3.2 Implementation

We use DeBERTa-v3 base and large as our language model [He et al., 2021a, He et al., 2021b]. We generate the visual imageries I_T and I_D by using an open-source DALL-E implementation [Ramesh et al., 2021, Dayma et al., 2021].[2] The number

[2]github.com/kuprel/min-dalle

Model	LM	*validation*	*test*
Vanilla Baseline	Base	79.84 ±2.23	-
+ Desc.	Base	86.39 ±1.05	83.58
+ Desc.	Large	<u>88.89</u> ±1.35	<u>85.74</u>
+ Desc. + Imag.	Large	**90.11** ±1.59	**87.16**

Table 5.1: Quantitative results on the labeled data using F1 as evaluation metric. The last two columns respectively show the average score over different *validation* splits, and the ensemble performance achieved on the *test* split.

of images per visual imagery K is set to 9. We extract visual imagery embeddings v_T and v_D using CLIP's ViT-L/14 as our visual encoder [Radford et al., 2021]. f_p is a single linear layer, and we randomly initialize its weights. We use Adam optimizer with weight decay [Kingma and Ba, 2015, Loshchilov and Hutter, 2018]. The learning rate is set to $5e^{-6}$ and $3e^{-6}$ for the experiments with DeBERTa-v3-base and DeBERTa-v3-large, respectively. We train our models for a maximum of 50 epochs using Tesla V100s and mixed precision. A typical experiment takes less than one hour with a batch size of 16. Due to the small dataset size, we perform multiple experiments and reserve a different portion of the labeled data for validation in each experiment. We report mean and standard deviation over all experiments, and use ensembling to evaluate our system on the test set.

5.4 Experimental Analysis

In this section, we report the quantitative results of our experiments and the preliminary outputs of our qualitative analysis.

5.4.1 Quantitative Results

Table 5.1 presents the quantitative results of our experiments as ablation studies. We perform several experiments in a curriculum, where each following experiment

activates a different feature (e.g. literal descriptions). We first implement a vanilla baseline using DeBERTa-v3-base, which lacks descriptions and imagery.

Using Literal Descriptions. In our first ablative analysis, we incorporate the literal term descriptions into the vanilla baseline described in Section 5.2.2. Integrating this supervision results in substantial performance improvement, i.e. $\approx$6.5 points using F1 as evaluation metric.

Larger Language Model. We implement the literal descriptions model using a larger language model which is the large architecture of the DeBERTa-v3 model. Using a bigger LM gives 2 points performance improvement.

Visual Imagery. We now report on the visual imagery model explained in Section 5.2.3. This model additionally uses two different visual embedding vectors, denoted as visual imageries, which are generated by a text-to-image model using terms and descriptions. By using this extra visual supervision, we obtain 1.22 and 1.42 F1 score increments in validation and testing phases. A paired t-test is applied to determine the significance of the results: We obtained a p-value of 0.032, which points out that this improvement is statistically significant ($p < 0.05$).

5.4.2 Qualitative Analysis

Figure 5.5 wraps up our qualitative analysis, where we share the collected descriptions and the generated visual imageries for some euphemistic terms. The first two examples show that if a term has a dominant literal meaning, the text-to-image V2I model produces images conveying the literal meaning instead of the figurative one. V2I can also produce visuals based upon individual word meanings as a consequence of being completely unconscious to the figurative meaning. This can be seen on the third example, where the model generates *lunch* images instead of vomiting for phrase '*lose one's lunch*'. Moreover, V2I can generate unrelated images for some terms as one can see on the *pro-life* and *able-body* examples. On the other hand, the text-to-image model V2I is well aware of some euphemism candidates as in the case with the last two examples. This phenomenon arises when the term has just one single meaning which is euphemistic.

In summary, a text-to-image model can be a complementary tool for analyzing figurative language: one can observe how models process these expressions. By looking at the produced images, we can recognize the terms with dominant literal meanings (e.g. *late*) or single euphemistic meaning (e.g. *lavatory*).

5.5 Related Work

5.5.1 Euphemisms.

Recently, euphemisms have attracted the attention of the natural language processing community. [Zhu et al., 2021] and [Zhu and Bhat, 2021] extract euphemistic phrases by using masked language modeling. A few work practices sentiment-oriented methods to recognize candidate euphemism phrases [Felt and Riloff, 2020, Gavidia et al., 2022, Lee et al., 2022]. Most notably, [Gavidia et al., 2022] replace PETs with their literal meanings and observe how the sentiment scores change. They demonstrate that using literal meanings produces higher scores for offensive speech and negative sentiment. Similarly, we also put literal meanings to use, but differently, by creating a textual input prompt. In this work, we also use the euphemism dataset they created.

5.5.2 Knowledge-augmented Language Understanding.

External knowledge[3] can be either unstructured (i.e. text) or structured (i.e. graph). To benefit from unstructured knowledge, a text retriever collects related entries from an external corpus [Karpukhin et al., 2020, Guu et al., 2020]. Conversely, structured knowledge integration may happen in two ways: explicit methods prefer to use knowledge in their input [Liu et al., 2020, Zhang et al., 2019], and implicit methods try to learn knowledge in their objective [Xiong et al., 2019, Shen et al., 2020]. Some exceptions [Yu et al., 2022a, Lv et al., 2020] combines both: they learn to predict graph embeddings and use these embeddings as input in their model concurrently. Similar to us, [Yu et al., 2022b, Xu et al., 2021b, Chakrabarty et al., 2021] also insert

[3]Please check [Zhu et al., 2022a] for a comprehensive review of the related literature.

descriptions into their textual inputs.

5.5.3 Visually-aided Language Understanding.

Several methods have been proposed to aid language learning with external visual knowledge. Most of these methods experiment on machine translation (MT). [Calixto et al., 2019] propose a latent variable model for multi-modal MT, to learn an association between an image and its target language description. [Long et al., 2021, Li et al., 2022d] first synthesize an image conditioned on the source sentence, then use both the source sentence and the synthesized image to produce translation. [Caglayan et al., 2020] obtain a lower latency in simultaneous MT by supplying visual context. Differently, Vokenization [Tan and Bansal, 2020] extends BERT [Devlin et al., 2019] by implementing visual token prediction objective to learn a mapping between tokens and associated images. Most relevantly, [Lu et al., 2022] improve text-only language understanding performance in low-resource settings by using generated imagination as visual supervision.

5.6 Conclusion

In this chapter, we described our two-stage method for the euphemism detection task. We first collected literal descriptions for PETs, inserted these descriptions into the model input, and showed that such linguistic supervision greatly boosts performance. We then supplied extra visual supervision using a text-to-image model, where we denote this kind of supervision as visual imageries. We achieved a statistically significant performance increase by using visual imageries in addition to the term descriptions. Our qualitative analysis on visual imageries also suggests that a text-to-image model can be a functional tool to break down how models process figures of speech.

Limitations. Due to working with a small dataset, we were able to collect descriptions for the PETs manually. Collecting these descriptions using an automatic retrieval system would be more sophisticated. We also did not perform detailed analyses of the results, which could help shed light on the contribution of each

model component. A critical limitation of this study is that we did not perform an experiment where we integrated extra visual supervision only without any extra linguistic supervision into the input prompt. Observing how much extra visual supervision could aid the model could be beneficial. That being said, extra visual supervision still aids the model, even when the answer becomes evident for the model with the literal description.

Term	**Description**	I_T	I_D
late	old person, elderly		
pass on	death, dying		
lose one's lunch	vomit, vomiting, throwing up		
pro-life	a person opposes abortion		
able-body	not disabled		
lavatory	restroom, toilet		
senior citizen	old person, elderly		

Figure 5.5: Examples of collected literal descriptions for euphemistic terms and their visual imageries.

Chapter 6

CONCLUSION

In this thesis, we investigated the recent challenges introduced in V&L learning: temporal reasoning and commonsense reasoning in problems bridging vision and language. To this end, we worked on four distinct problems bearing these challenges.

We first practiced the localization stage in Chapter 2 before tackling temporal and commonsense reasoning. Our experiments on two separate tasks revealed that conditioning language on the bottom-up/contracting visual processing branch results in more robust low-level concept grounding.

In Chapter 3 and Chapter 4, we proposed two different types of benchmarks to measure the temporal reasoning and commonsense reasoning capabilities of the pretrained video-language models. Our experimental analyses on these benchmarks made it apparent that the current video-language models are inadequate in terms of temporal and commonsense reasoning, falling far behind humans.

In Chapter 3, we showed that the existing video-language models then cannot capture the physical interactions/events happening in videos, while an oracle baseline accessing these events as a chain-of-thought prompt demonstrates superior performance – on par with human beings. This finding suggests that devising a model that proficiently transcribes frame sequences into event sequences in written form could be a good starting point to excel in this task.

In Chapter 4, we developed two sorts of foiling tasks for pretrained video-language models by prioritizing temporal and commonsense reasoning. In the first task, we presented an extreme temporal reasoning challenge for video-language models: counting the repeated actions in given videos. We showed that the current models act like chance-based random baselines in this task. In the second task, we came up with a task that assesses the compositionality capacities of the models, which

involves recognizing events less likely to occur in the physical world. Our experiments revealed that the current models fail to identify actions for 20% of examples in the rare action recognition tests, indicating that their physical commonsense reasoning skills fail within atypical contexts.

We focused on a figurative language understanding task in Chapter 5, called euphemism detection. Our experiments revealed that incorporating both textual and visual knowledge into the reasoning benefits low-resource commonsense reasoning scenarios.

Overall, these four studies emphasize two outcomes: First, the current state-of-the-art video-language models fail to perform adequately on the current benchmarks that require temporal reasoning. This finding should motivate the community to advance by devising better video-language modeling methodologies to thrive in spatio-temporal language grounding benchmarks. Second, extra knowledge could benefit commonsense reasoning tasks. We show that extra textual information about a given context aids reasoning in Chapter 3 and Chapter 5. Furthermore, in Chapter 5, we also observed a significant increase when we enhanced input with extra visual knowledge illustrating figurative terms and their literal meanings. This finding encourages us to fuse visual and other modalities in text-only commonsense reasoning problems to supply more knowledge about the world.

6.1 Future Directions

In this part of the thesis, we briefly summarize some potential directions for the future work based on our findings.

6.1.1 Creating Proficient Video-Language Models

Our findings in Chapter 3 and Chapter 4 motivate us to discover more about video-language model development. The insufficiency of the existing video-language models could be due to the following two reasons.

The first reason could be the overall low quality of pretraining datasets. For instance, the HowTo100M dataset [Miech et al., 2019] does not contain captions:

the captions for this particular dataset are created by employing an Automatic Speech Recognition (ASR) system. Apart from the errors that an ASR system could introduce, the actual problem is that the speech might not be a suitable signal to relate spatio-temporal events directly. The recent WebVid2M and WebVid10M datasets [Bain et al., 2021] did not have this drawback as they include ground truth captions. However, their data collection process is similar to CC3M [Sharma et al., 2018], which results in noisy captions. [Li et al., 2022b] shows that longer pretraining with this kind of noisy web data does not boost performance. Recent studies [Gunasekar et al., 2023, Li et al., 2023d] show that pretraining language models with clean data is vital to performing well on various problems, including code generation and common sense reasoning.

Regarding these findings, the community should pursue directions employing pretraining with clean data. In this manner, VATEX [Wang et al., 2019d] could be a prominent resource. Afterwards, people can implement pretraining strategies similar to BLIP [Li et al., 2022b, Li et al., 2023c] and proceed to multi-task instruction tuning in video-language model training similar to InstructBLIP [Dai et al., 2023] and T0 [Sanh et al., 2021].

The second reason could be insufficient spatio-temporal modeling. Due to the computational constraints, most video-language models implement a downsampling mechanism to represent input videos: Even the most recent models (see Chapter 4) process just 4 frames as video input. Moreover, these models use an image encoder solely pretrained on static images [Bao et al., 2021] or an outdated video encoder [Xie et al., 2018] as their video processing backbone. So, the first step should be replacing these unsuitable components with recent video representation learning pipelines such as [Wang et al., 2023a, Sun et al., 2023]. If this process leads to better temporal grounding capabilities, the second step could involve devising better self-supervised video representation learning methodologies. That being said, these strategies might not be enough to thrive in benchmarks like CRAFT, due to the complex dynamic scenes. One might therefore need to experiment with object-centric solutions like [Singh et al., 2022].

6.1.2 Domain Expert Models

In this thesis, we tackled problems requiring commonsense knowledge, i.e., the implicit general knowledge about the world we are living in. This sort of tasks requires no additional knowledge from a regular person. A more challenging direction could be to develop domain expert models, which have specialized knowledge in a particular domain such as mathematics [Lu et al., 2023, Petersen et al., 2023], medicine [Labrak et al., 2023, Xia et al., 2022, Elfrink et al., 2023], and law [Chalkidis et al., 2023, Zhang et al., 2023]. In particular, developing multimodal language models for the medical domain remains a promising direction to follow upon this thesis, since medical data can often comprise visual modality such as radiology images [Dalla Serra et al., 2022, Hou et al., 2023]. This area also caught the attention of the community, which resulted in the emergence of many different medical vision-language models recently [Wang et al., 2022c, Wu et al., 2023, Singhal et al., 2023, Moor et al., 2023, Li et al., 2023b].

6.1.3 Multilingual and Cross-Lingual Learning

In this thesis, we did not investigate one aspect of vision-language learning: multilingual and/or cross-lingual learning. This is a missing piece in unimodal language learning as well: The community showed immense interest in English, leaving the other languages less researched than English. We are aware of the recent efforts toward this direction both in unimodal [Xue et al., 2021, Lin et al., 2021, Emelin and Sennrich, 2021, Safaya et al., 2022, Pikuliak et al., 2022] and multimodal domains [Liu et al., 2021a, Nooralahzadeh and Sennrich, 2023, Bugliarello et al., 2022, Verma et al., 2023, Gan et al., 2023, Chen et al., 2023]. Our future work involves devising solutions that transfer knowledge from high-resource languages to low-resource languages in a data-efficient manner, by making use of small-scale datasets such as [Elliott et al., 2016, Unal et al., 2016].

BIBLIOGRAPHY

[Akbari et al., 2021] Akbari, H., Yuan, L., Qian, R., Chuang, W.-H., Chang, S.-F., Cui, Y., and Gong, B. (2021). Vatt: Transformers for multimodal self-supervised learning from raw video, audio and text. *Advances in Neural Information Processing Systems*, 34:24206–24221.

[Allen et al., 2020] Allen, K. R., Smith, K. A., and Tenenbaum, J. B. (2020). Rapid trial-and-error learning with simulation supports flexible tool use and physical reasoning. *arXiv preprint arXiv:1907.09620*.

[Antol et al., 2015] Antol, S., Agrawal, A., Lu, J., Mitchell, M., Batra, D., Lawrence Zitnick, C., and Parikh, D. (2015). VQA: Visual question answering. In *ICCV*, pages 2425–2433.

[Aroca-Ouellette et al., 2020] Aroca-Ouellette, S., Paik, C., Roncone, A., and Kann, K. (2020). Prost: Physical reasoning of objects through space and time. In *ACL-Findinfs*.

[Ateş, 2021] Ateş, T. (2021). Towards understanding intuitive physics with language and vision.

[Ates et al., 2022] Ates, T., Ateşoğlu, M., Yiğit, Ç., Kesen, I., Kobas, M., Erdem, E., Erdem, A., Goksun, T., and Yuret, D. (2022). CRAFT: A benchmark for causal reasoning about forces and inTeractions. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2602–2627, Dublin, Ireland. Association for Computational Linguistics.

[Bagad et al., 2023] Bagad, P., Tapaswi, M., and Snoek, C. G. M. (2023). Test of Time: Instilling Video-Language Models with a Sense of Time. In *CVPR*.

[Bahdanau et al., 2014] Bahdanau, D., Cho, K., and Bengio, Y. (2014). *Neural Machine Translation by Jointly Learning to Align and Translate*.

[Bahng et al., 2018] Bahng, H., Yoo, S., Cho, W., Park, D. K., Wu, Z., Ma, X., and Choo, J. (2018). Coloring with words: Guiding image colorization through text-based palette generation. In *ECCV*, pages 431–447.

[Bain et al., 2021] Bain, M., Nagrani, A., Varol, G., and Zisserman, A. (2021). 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.

[Bakhtin et al., 2019] Bakhtin, A., van der Maaten, L., Johnson, J., Gustafson, L., and Girshick, R. (2019). Phyre: A new benchmark for physical reasoning. In *NeurIPS*, pages 5082–5093.

[Bao et al., 2021] Bao, H., Dong, L., and Wei, F. (2021). Beit: Bert pre-training of image transformers. *ArXiv*, abs/2106.08254.

[Baradel et al., 2020] Baradel, F., Neverova, N., Mille, J., Mori, G., and Wolf, C. (2020). CoPhy: Counterfactual learning of physical dynamics. In *ICLR*.

[Bernardi et al., 2016] Bernardi, R., Cakici, R., Elliott, D., Erdem, A., Erdem, E., Ikizler-Cinbis, N., Keller, F., Muscat, A., and Plank, B. (2016). Automatic description generation from images: A survey of models, datasets, and evaluation measures. *Journal of Artificial Intelligence Research*, 55:409–442.

[Bertasius et al., 2021] Bertasius, G., Wang, H., and Torresani, L. (2021). Is space-time attention all you need for video understanding? In *ICML*, volume 2, page 4.

[Bisk et al., 2020a] Bisk, Y., Holtzman, A., Thomason, J., Andreas, J., Bengio, Y., Chai, J., Lapata, M., Lazaridou, A., May, J., Nisnevich, A., Pinto, N., and Turian, J. (2020a). Experience grounds language. In *Proceedings of the 2020 Conference*

on Empirical Methods in Natural Language Processing (EMNLP), pages 8718–8735, Online. Association for Computational Linguistics.

[Bisk et al., 2020b] Bisk, Y., Zellers, R., Gao, J., Choi, Y., et al. (2020b). Piqa: Reasoning about physical commonsense in natural language. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 7432–7439.

[Bitton et al., 2021] Bitton, Y., Stanovsky, G., Schwartz, R., and Elhadad, M. (2021). Automatic generation of contrast sets from scene graphs: Probing the compositional consistency of GQA. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 94–105, Online. Association for Computational Linguistics.

[Bojar et al., 2014] Bojar, O., Buck, C., Federmann, C., Haddow, B., Koehn, P., Leveling, J., Monz, C., Pecina, P., Post, M., Saint-Amand, H., Soricut, R., Specia, L., and Tamchyna, A. (2014). Findings of the 2014 workshop on statistical machine translation. In *Proceedings of the Ninth Workshop on Statistical Machine Translation*, pages 12–58, Baltimore, Maryland, USA. Association for Computational Linguistics.

[Boutonnet and Lupyan, 2015] Boutonnet, B. and Lupyan, G. (2015). Words jump-start vision: A label advantage in object recognition. *Journal of Neuroscience*, 35(25):9329–9335.

[Buch et al., 2022] Buch, S., Eyzaguirre, C., Gaidon, A., Wu, J., Fei-Fei, L., and Niebles, J. C. (2022). Revisiting the" video" in video-language understanding. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2917–2927.

[Bugliarello et al., 2022] Bugliarello, E., Liu, F., Pfeiffer, J., Reddy, S., Elliott, D., Ponti, E. M., and Vulić, I. (2022). Iglue: A benchmark for transfer learning across modalities, tasks, and languages. In *International Conference on Machine Learning*, pages 2370–2392. PMLR.

[Bugliarello et al., 2023] Bugliarello, E., Sartran, L., Agrawal, A., Hendricks, L. A., and Nematzadeh, A. (2023). Measuring progress in fine-grained vision-and-language understanding. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1559–1582, Toronto, Canada. Association for Computational Linguistics.

[Caglayan et al., 2020] Caglayan, O., Ive, J., Haralampieva, V., Madhyastha, P., Barrault, L., and Specia, L. (2020). Simultaneous machine translation with visual context. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2350–2361, Online. Association for Computational Linguistics.

[Calixto et al., 2019] Calixto, I., Rios, M., and Aziz, W. (2019). Latent variable model for multi-modal translation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6392–6405, Florence, Italy. Association for Computational Linguistics.

[Can, 2021] Can, O. A. (2021). *Cognitively-Inspired Deep Learning Approaches for Grounded Language Learning*. PhD thesis, Koç University.

[Carion et al., 2020] Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., and Zagoruyko, S. (2020). End-to-end object detection with transformers. In *European conference on computer vision*, pages 213–229. Springer.

[Chakrabarty et al., 2021] Chakrabarty, T., Ghosh, D., Poliak, A., and Muresan, S. (2021). Figurative Language in Recognizing Textual Entailment. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 3354–3361, Online. Association for Computational Linguistics.

[Chalkidis et al., 2023] Chalkidis, I., Garneau, N., Goanta, C., Katz, D., and Søgaard, A. (2023). LeXFiles and LegalLAMA: Facilitating English multinational legal language model development. In *Proceedings of the 61st Annual Meeting of*

the Association for Computational Linguistics (Volume 1: Long Papers), pages 15513–15535, Toronto, Canada. Association for Computational Linguistics.

[Chandu et al., 2021] Chandu, K. R., Bisk, Y., and Black, A. W. (2021). Grounding 'grounding' in NLP. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 4283–4305, Online. Association for Computational Linguistics.

[Changpinyo et al., 2021] Changpinyo, S., Sharma, P., Ding, N., and Soricut, R. (2021). Conceptual 12M: Pushing web-scale image-text pre-training to recognize long-tail visual concepts. In *CVPR*.

[Chen et al., 2019a] Chen, D.-J., Jia, S., Lo, Y.-C., Chen, H.-T., and Liu, T.-L. (2019a). See-through-text grouping for referring image segmentation. In *ICCV*, pages 7454–7463.

[Chen et al., 2023] Chen, G., Hou, L., Chen, Y., Dai, W., Shang, L., Jiang, X., Liu, Q., Pan, J., and Wang, W. (2023). mCLIP: Multilingual CLIP via cross-lingual transfer. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13028–13043, Toronto, Canada. Association for Computational Linguistics.

[Chen et al., 2018a] Chen, J., Shen, Y., Gao, J., Liu, J., and Liu, X. (2018a). Language-based image editing with recurrent attentive models. In *CVPR*, pages 8721–8729.

[Chen et al., 2018b] Chen, L.-C., Zhu, Y., Papandreou, G., Schroff, F., and Adam, H. (2018b). Encoder-decoder with atrous separable convolution for semantic image segmentation. In *ECCV*, pages 801–818.

[Chen et al., 2019b] Chen, Y.-W., Tsai, Y.-H., Wang, T., Lin, Y.-Y., and Yang, M.-H. (2019b). Referring Expression Object Segmentation with Caption-Aware Consistency. In *BMVC*.

[Chen et al., 2021] Chen, Z., Mao, J., Wu, J., Wong, K.-Y. K., Tenenbaum, J. B., and Gan, C. (2021). Grounding physical concepts of objects and events through dynamic visual reasoning. In *ICLR*.

[Cheng et al., 2023] Cheng, F., Wang, X., Lei, J., Crandall, D., Bansal, M., and Bertasius, G. (2023). Vindlu: A recipe for effective video-and-language pretraining. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10739–10750.

[Chung et al., 2014] Chung, J., Gulcehre, C., Cho, K., and Bengio, Y. (2014). Empirical evaluation of gated recurrent neural networks on sequence modeling. In *NIPS 2014 Workshop on Deep Learning, December 2014*.

[Cirik et al., 2018] Cirik, V., Berg-Kirkpatrick, T., and Morency, L.-P. (2018). Using syntax to ground referring expressions in natural images. In *AAAI*.

[Dai et al., 2023] Dai, W., Li, J., Li, D., Tiong, A. M. H., Zhao, J., Wang, W., Li, B., Fung, P., and Hoi, S. (2023). Instructblip: Towards general-purpose vision-language models with instruction tuning.

[Dalla Serra et al., 2022] Dalla Serra, F., Clackett, W., MacKinnon, H., Wang, C., Deligianni, F., Dalton, J., and O'Neil, A. Q. (2022). Multimodal generation of radiology reports using knowledge-grounded extraction of entities and relations. In *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 615–624, Online only. Association for Computational Linguistics.

[Das et al., 2017a] Das, A., Kottur, S., Gupta, K., Singh, A., Yadav, D., Moura, J. M., Parikh, D., and Batra, D. (2017a). Visual Dialog. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[Das et al., 2017b] Das, A., Kottur, S., Moura, J. M., Lee, S., and Batra, D. (2017b). Learning cooperative visual dialog agents with deep reinforcement learning. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*.

[Dayma et al., 2021] Dayma, B., Patil, S., Cuenca, P., Saifullah, K., Abraham, T., Le Khac, P., Melas, L., and Ghosh, R. (2021). Dall·e mini.

[De Vries et al., 2017] De Vries, H., Strub, F., Mary, J., Larochelle, H., Pietquin, O., and Courville, A. C. (2017). Modulating early visual processing by language. In *NeurIPS*, pages 6594–6604.

[Deng et al., 2009a] Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. (2009a). Imagenet: A large-scale hierarchical image database. In *CVPR*, pages 248–255.

[Deng et al., 2009b] Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. (2009b). Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee.

[Deng et al., 2021] Deng, J., Yang, Z., Chen, T., Zhou, W., and Li, H. (2021). Transvg: End-to-end visual grounding with transformers. In *ICCV*, pages 1769–1779.

[Dessalegn and Landau, 2013] Dessalegn, B. and Landau, B. (2013). Interaction between language and vision: It's momentary, abstract, and it develops. *Cognition*, 127(3):331–344.

[Devlin et al., 2019] Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

[Ding et al., 2020] Ding, D., Hill, F., Santoro, A., and Botvinick, M. (2020). Object-based attention for spatio-temporal reasoning: Outperforming neuro-symbolic models with flexible distributed architectures. *arXiv preprint arXiv:2012.08508*.

[Ding et al., 2021a] Ding, H., Liu, C., Wang, S., and Jiang, X. (2021a). Vision-language transformer and query generation for referring segmentation. In *ICCV*.

[Ding et al., 2021b] Ding, M., Chen, Z., Du, T., Luo, P., Tenenbaum, J. B., and Gan, C. (2021b). Dynamic visual reasoning by learning differentiable physics models from video and language. In *Advances In Neural Information Processing Systems*.

[Dosovitskiy et al., 2020] Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N. (2020). An image is worth 16x16 words: Transformers for image recognition at scale. *ArXiv*, abs/2010.11929.

[Dosovitskiy et al., 2021] Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N. (2021). An image is worth 16x16 words: Transformers for image recognition at scale. *ICLR*.

[Elfrink et al., 2023] Elfrink, A., Vagliano, I., Abu-Hanna, A., and Calixto, I. (2023). Soft-prompt tuning to predict lung cancer using primary care free-text dutch medical notes. In Juarez, J. M., Marcos, M., Stiglic, G., and Tucker, A., editors, *Artificial Intelligence in Medicine*, pages 193–198, Cham. Springer Nature Switzerland.

[Elliott et al., 2016] Elliott, D., Frank, S., Sima'an, K., and Specia, L. (2016). Multi30K: Multilingual English-German image descriptions. In *Proceedings of the 5th Workshop on Vision and Language*, pages 70–74, Berlin, Germany. Association for Computational Linguistics.

[Emelin and Sennrich, 2021] Emelin, D. and Sennrich, R. (2021). Wino-X: Multilingual Winograd schemas for commonsense reasoning and coreference resolution. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 8517–8532, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

[Fan et al., 2019] Fan, A., Jernite, Y., Perez, E., Grangier, D., Weston, J., and Auli, M. (2019). ELI5: Long form question answering. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3558–3567, Florence, Italy. Association for Computational Linguistics.

[Felt and Riloff, 2020] Felt, C. and Riloff, E. (2020). Recognizing euphemisms and dysphemisms using sentiment analysis. In *Proceedings of the Second Workshop on Figurative Language Processing*, pages 136–145, Online. Association for Computational Linguistics.

[Feng et al., 2021] Feng, G., Hu, Z., Zhang, L., and Lu, H. (2021). Encoder fusion network with co-attention embedding for referring image segmentation. In *CVPR*, pages 15506–15515.

[Fu et al., 2021] Fu, T.-J., Li, L., Gan, Z., Lin, K., Wang, W. Y., Wang, L., and Liu, Z. (2021). Violet: End-to-end video-language transformers with masked visual-token modeling. *arXiv preprint arXiv:2111.12681*.

[Gan et al., 2023] Gan, T., Wang, Q., Dong, X., Ren, X., Nie, L., and Guo, Q. (2023). Cnvid-3.5m: Build, filter, and pre-train the large-scale public chinese video-text dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14815–14824.

[Gao et al., 2018] Gao, P., Li, H., Li, S., Lu, P., Li, Y., Hoi, S. C., and Wang, X. (2018). Question-guided hybrid convolution for visual question answering. In *ECCV*, pages 469–485.

[Gavidia et al., 2022] Gavidia, M., Lee, P., Feldman, A., and Peng, J. (2022). CATs are Fuzzy PETs: A Corpus and Analysis of Potentially Euphemistic Terms. *arXiv preprint arXiv:2205.02728*.

[Gavrilyuk et al., 2018] Gavrilyuk, K., Ghodrati, A., Li, Z., and Snoek, C. G. (2018). Actor and action video segmentation from a sentence. In *CVPR*, pages 5958–5966.

[Ge et al., 2022] Ge, Y., Ge, Y., Liu, X., Li, D., Shan, Y., Qie, X., and Luo, P. (2022). Bridging video-text retrieval with multiple choice questions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16167–16176.

[Gillick et al., 2016] Gillick, D., Brunk, C., Vinyals, O., and Subramanya, A. (2016). Multilingual language processing from bytes. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1296–1306, San Diego, California. Association for Computational Linguistics.

[Girdhar and Ramanan, 2020] Girdhar, R. and Ramanan, D. (2020). CATER: A diagnostic dataset for compositional actions and temporal reasoning. In *ICLR*.

[Girshick et al., 2014] Girshick, R., Donahue, J., Darrell, T., and Malik, J. (2014). Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. *CVPR*.

[Gokhale et al., 2020] Gokhale, T., Banerjee, P., Baral, C., and Yang, Y. (2020). MUTANT: A training paradigm for out-of-distribution generalization in visual question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 878–892, Online. Association for Computational Linguistics.

[Goodfellow et al., 2014] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative

adversarial nets. In Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N., and Weinberger, K. Q., editors, *NeurIPS*, volume 27.

[Goyal et al., 2017a] Goyal, Y., Khot, T., Summers-Stay, D., Batra, D., and Parikh, D. (2017a). Making the V in VQA matter: Elevating the role of image understanding in Visual Question Answering. In *Conference on Computer Vision and Pattern Recognition (CVPR)*.

[Goyal et al., 2017b] Goyal, Y., Khot, T., Summers-Stay, D., Batra, D., and Parikh, D. (2017b). Making the V in VQA matter: Elevating the role of image understanding in visual question answering. In *CVPR*, pages 6904–6913.

[Gregor et al., 2015] Gregor, K., Danihelka, I., Graves, A., Rezende, D., and Wierstra, D. (2015). Draw: A recurrent neural network for image generation. In Bach, F. and Blei, D., editors, *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 1462–1471, Lille, France. PMLR.

[Gunasekar et al., 2023] Gunasekar, S., Zhang, Y., Aneja, J., Mendes, C. C. T., Giorno, A. D., Gopi, S., Javaheripi, M., Kauffmann, P., de Rosa, G., Saarikivi, O., Salim, A., Shah, S., Behl, H. S., Wang, X., Bubeck, S., Eldan, R., Kalai, A. T., Lee, Y. T., and Li, Y. (2023). Textbooks are all you need.

[Guu et al., 2020] Guu, K., Lee, K., Tung, Z., Pasupat, P., and Chang, M. (2020). Retrieval augmented language model pre-training. In *International Conference on Machine Learning*, pages 3929–3938. PMLR.

[He et al., 2017] He, K., Gkioxari, G., Dollar, P., and Girshick, R. (2017). Mask R-CNN. *ICCV*.

[He et al., 2016a] He, K., Zhang, X., Ren, S., and Sun, J. (2016a). Deep residual learning for image recognition. In *CVPR*, pages 770–778.

[He et al., 2016b] He, K., Zhang, X., Ren, S., and Sun, J. (2016b). Deep residual learning for image recognition. In *CVPR*, pages 770–778.

[He et al., 2021a] He, P., Gao, J., and Chen, W. (2021a). Debertav3: Improving deberta using electra-style pre-training with gradient-disentangled embedding sharing.

[He et al., 2021b] He, P., Liu, X., Gao, J., and Chen, W. (2021b). Deberta: Decoding-enhanced bert with disentangled attention. In *International Conference on Learning Representations*.

[Hendricks and Nematzadeh, 2021] Hendricks, L. A. and Nematzadeh, A. (2021). Probing image-language transformers for verb understanding. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 3635–3644, Online. Association for Computational Linguistics.

[Hendricks et al., 2017] Hendricks, L. A., Wang, O., Shechtman, E., Sivic, J., Darrell, T., and Russell, B. (2017). Localizing moments in video with natural language. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*.

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

[Hochreiter and Schmidhuber, 1997b] Hochreiter, S. and Schmidhuber, J. (1997b). Long short-term memory. *Neural Comput.*, 9(8):1735–1780.

[Holder, 2008] Holder, R. W. (2008). *Dictionary of euphemisms*. Oxford University Press.

[Hong et al., 2021] Hong, X., Lan, Y., Pang, L., Guo, J., and Cheng, X. (2021). Transformation driven visual reasoning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6903–6912.

[Hou et al., 2023] Hou, W., Xu, K., Cheng, Y., Li, W., and Liu, J. (2023). ORGAN: Observation-guided radiology report generation via tree reasoning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8108–8122, Toronto, Canada. Association for Computational Linguistics.

[Hu et al., 2017] Hu, R., Rohrbach, M., Andreas, J., Darrell, T., and Saenko, K. (2017). Modeling Relationships in Referential Expressions with Compositional Modular Networks. *CVPR*.

[Hu et al., 2016] Hu, R., Rohrbach, M., and Darrell, T. (2016). Segmentation from Natural Language Expressions. *Lecture Notes in Computer Science*, pages 108–124.

[Hu et al., 2020] Hu, Z., Feng, G., Sun, J., Zhang, L., and Lu, H. (2020). Bi-Directional Relationship Inferring Network for Referring Image Segmentation. In *CVPR*, pages 4424–4433.

[Huang et al., 2020] Huang, S., Hui, T., Liu, S., Li, G., Wei, Y., Han, J., Liu, L., and Li, B. (2020). Referring image segmentation via cross-modal progressive comprehension. In *CVPR*, pages 10488–10497.

[Hudson and Manning, 2018] Hudson, D. A. and Manning, C. D. (2018). Compositional attention networks for machine reasoning. In *ICLR*.

[Hui et al., 2020] Hui, T., Liu, S., Huang, S., Li, G., Yu, S., Zhang, F., and Han, J. (2020). Linguistic structure guided context modeling for referring image segmentation. In *ECCV*, pages 59–75. Springer.

[Ioffe and Szegedy, 2015] Ioffe, S. and Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *ICML*, pages 448–456. PMLR.

[Janner et al., 2019] Janner, M., Levine, S., Freeman, W. T., Tenenbaum, J. B., Finn, C., and Wu, J. (2019). Reasoning about physical interactions with object-oriented prediction and planning. In *ICLR*.

[Jing et al., 2021] Jing, Y., Kong, T., Wang, W., Wang, L., Li, L., and Tan, T. (2021). Locate then segment: A strong pipeline for referring image segmentation. In *CVPR*, pages 9858–9867.

[Johnson et al., 2017] Johnson, J., Hariharan, B., van der Maaten, L., Fei-Fei, L., Lawrence Zitnick, C., and Girshick, R. (2017). Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. In *CVPR*, pages 2901–2910.

[Karpukhin et al., 2020] Karpukhin, V., Oguz, B., Min, S., Lewis, P., Wu, L., Edunov, S., Chen, D., and Yih, W.-t. (2020). Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781, Online. Association for Computational Linguistics.

[Kazemzadeh et al., 2014] Kazemzadeh, S., Ordonez, V., Matten, M., and Berg, T. (2014). Referitgame: Referring to objects in photographs of natural scenes. In *EMNLP*, pages 787–798.

[Kesen et al., 2022a] Kesen, I., Can, O. A., Erdem, E., Erdem, A., and Yüret, D. (2022a). Modulating bottom-up and top-down visual processing via language-conditional filters. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 4610–4620.

[Kesen et al., 2022b] Kesen, I., Erdem, A., Erdem, E., and Calixto, I. (2022b). Detecting euphemisms with literal descriptions and visual imagery. In *Proceedings of the 3rd Workshop on Figurative Language Processing (FLP)*, pages 61–67, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

[Kiela and Clark, 2015] Kiela, D. and Clark, S. (2015). Multi- and cross-modal semantics beyond vision: Grounding in auditory perception. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2461–2470, Lisbon, Portugal. Association for Computational Linguistics.

[Kim et al., 2019] Kim, H., Jhoo, H. Y., Park, E., and Yoo, S. (2019). Tag2pix: Line art colorization using text tag with secat and changing loss. In *ICCV*, pages 9056–9065.

[Kingma and Ba, 2014a] Kingma, D. P. and Ba, J. (2014a). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

[Kingma and Ba, 2014b] Kingma, D. P. and Ba, J. (2014b). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

[Kingma and Ba, 2015] Kingma, D. P. and Ba, J. (2015). Adam: A method for stochastic optimization. In *ICLR (Poster)*.

[Krizhevsky et al., 2012] Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In Pereira, F., Burges, C., Bottou, L., and Weinberger, K., editors, *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc.

[Kuehne et al., 2011] Kuehne, H., Jhuang, H., Garrote, E., Poggio, T., and Serre, T. (2011). Hmdb: a large video database for human motion recognition. In *2011 International conference on computer vision*, pages 2556–2563. IEEE.

[Labrak et al., 2023] Labrak, Y., Bazoge, A., Dufour, R., Rouvier, M., Morin, E., Daille, B., and Gourraud, P.-A. (2023). DrBERT: A robust pre-trained model in French for biomedical and clinical domains. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16207–16221, Toronto, Canada. Association for Computational Linguistics.

[Lan et al., 2020] Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P., and Soricut, R. (2020). Albert: A lite bert for self-supervised learning of language representations. In *International Conference on Learning Representations*.

[LeCun et al., 1989] LeCun, Y., Boser, B., Denker, J., Henderson, D., Howard, R., Hubbard, W., and Jackel, L. (1989). Handwritten digit recognition with a back-propagation network. *Advances in neural information processing systems*, 2.

[Lee et al., 2022] Lee, P., Gavidia, M., Feldman, A., and Peng, J. (2022). Searching for PETs: Using Distributional and Sentiment-Based Methods to Find Potentially Euphemistic Terms. In *Proceedings of the Second Workshop on Understanding Implicit and Underspecified Language*, pages 22–32, Seattle, USA. Association for Computational Linguistics.

[Lei et al., 2023] Lei, J., Berg, T., and Bansal, M. (2023). Revealing single frame bias for video-and-language learning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 487–507, Toronto, Canada. Association for Computational Linguistics.

[Lei et al., 2022] Lei, J., Berg, T. L., and Bansal, M. (2022). Revealing single frame bias for video-and-language learning. *arXiv preprint arXiv:2206.03428*.

[Lei et al., 2021] Lei, J., Li, L., Zhou, L., Gan, Z., Berg, T. L., Bansal, M., and Liu, J. (2021). Less is more: Clipbert for video-and-language learning via sparse sampling. In *CVPR*.

[Lei et al., 2018] Lei, J., Yu, L., Bansal, M., and Berg, T. L. (2018). TVQA: Localized, compositional video question answering. In *EMNLP*.

[Lei et al., 2020a] Lei, J., Yu, L., Berg, T., and Bansal, M. (2020a). What is more likely to happen next? video-and-language future event prediction. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8769–8784, Online. Association for Computational Linguistics.

[Lei et al., 2020b] Lei, J., Yu, L., Berg, T. L., and Bansal, M. (2020b). TVQA+: Spatio-temporal grounding for video question answering. In *ACL*.

[Lerer et al., 2016] Lerer, A., Gross, S., and Fergus, R. (2016). Learning physical intuition of block towers by example. In *ICML*.

[Lewis et al., 2020] Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., Stoyanov, V., and Zettlemoyer, L. (2020). BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

[Li et al., 2023a] Li, B., Wang, R., Wang, G., Ge, Y., Ge, Y., and Shan, Y. (2023a). SEED-Bench: Benchmarking Multimodal LLMs with Generative Comprehension. arXiv:2307.16125 [cs].

[Li et al., 2023b] Li, C., Wong, C., Zhang, S., Usuyama, N., Liu, H., Yang, J., Naumann, T., Poon, H., and Gao, J. (2023b). Llava-med: Training a large language-and-vision assistant for biomedicine in one day.

[Li et al., 2022a] Li, D., Li, J., Li, H., Niebles, J. C., and Hoi, S. C. (2022a). Align and prompt: Video-and-language pre-training with entity prompts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4953–4963.

[Li et al., 2023c] Li, J., Li, D., Savarese, S., and Hoi, S. (2023c). Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. *ArXiv*, abs/2301.12597.

[Li et al., 2022b] Li, J., Li, D., Xiong, C., and Hoi, S. C. H. (2022b). Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International Conference on Machine Learning*.

[Li et al., 2021a] Li, J., Selvaraju, R. R., Gotmare, A. D., Joty, S., Xiong, C., and Hoi, S. (2021a). Align before fuse: Vision and language representation learning with momentum distillation. In *NeurIPS*.

[Li et al., 2020] Li, L., Chen, Y.-C., Cheng, Y., Gan, Z., Yu, L., and Liu, J. (2020). HERO: Hierarchical encoder for Video+Language omni-representation pre-training. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2046–2065, Online. Association for Computational Linguistics.

[Li et al., 2022c] Li, L., Gan, Z., Lin, K., Lin, C.-C., Liu, Z., Liu, C., and Wang, L. (2022c). Lavender: Unifying video-language understanding as masked language modeling. *arXiv preprint arXiv:2206.07160*.

[Li et al., 2021b] Li, L., Lei, J., Gan, Z., Yu, L., Chen, Y.-C., Pillai, R., Cheng, Y., Zhou, L., Wang, X. E., Wang, W. Y., et al. (2021b). Value: A multi-task benchmark for video-and-language understanding evaluation. In *35th Conference on Neural Information Processing Systems (NeurIPS 2021) Track on Datasets and Benchmarks*.

[Li et al., 2023d] Li, Y., Bubeck, S., Eldan, R., Giorno, A. D., Gunasekar, S., and Lee, Y. T. (2023d). Textbooks are all you need ii: phi-1.5 technical report.

[Li et al., 2022d] Li, Y., Panda, R., Kim, Y., Chen, C.-F. R., Feris, R. S., Cox, D., and Vasconcelos, N. (2022d). VALHALLA: Visual Hallucination for Machine Translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5216–5226.

[Li et al., 2019] Li, Y., Wu, J., Tedrake, R., Tenenbaum, J. B., and Torralba, A. (2019). Learning particle dynamics for manipulating rigid bodies, deformable objects, and fluids. *ICLR*.

[Li et al., 2017] Li, Z., Tao, R., Gavves, E., Snoek, C. G., and Smeulders, A. W. (2017). Tracking by natural language specification. In *CVPR*, pages 6495–6503.

[Liang et al., 2022] Liang, C., Wang, W., Zhou, T., and Yang, Y. (2022). Visual abductive reasoning. In *IEEE/CVF International Conference on Computer Vision and Pattern Recognition (CVPR)*.

[Liao et al., 2020] Liao, Y., Liu, S., Li, G., Wang, F., Chen, Y., Qian, C., and Li, B. (2020). A real-time cross-modality correlation filtering method for referring expression comprehension. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

[Lin et al., 2020a] Lin, B. Y., Lee, S., Khanna, R., and Ren, X. (2020a). Birds have four legs?! NumerSense: Probing Numerical Commonsense Knowledge of Pre-Trained Language Models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6862–6868, Online. Association for Computational Linguistics.

[Lin et al., 2021] Lin, B. Y., Lee, S., Qiao, X., and Ren, X. (2021). Common sense beyond English: Evaluating and improving multilingual language models for commonsense reasoning. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1274–1287, Online. Association for Computational Linguistics.

[Lin et al., 2014a] Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. (2014a). Microsoft coco: Common objects in context. In *ECCV*, pages 740–755. Springer.

[Lin et al., 2014b] Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. (2014b). Microsoft coco: Common objects in context. In *ECCV*, pages 740–755.

[Lin et al., 2022] Lin, Y.-B., Lei, J., Bansal, M., and Bertasius, G. (2022). Eclipse: Efficient long-range video retrieval using sight and sound. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXXIV*, pages 413–430. Springer.

[Lin et al., 2020b] Lin, Z., Wu, Y.-F., Peri, S., Fu, B., Jiang, J., and Ahn, S. (2020b). Improving generative imagination in object-centric world models. In *International Conference on Machine Learning*, pages 6140–6149. PMLR.

[Liu et al., 2017] Liu, C., Lin, Z. L., Shen, X., Yang, J., Lu, X., and Yuille, A. L. (2017). Recurrent Multimodal Interaction for Referring Image Segmentation. *ICCV*, pages 1280–1289.

[Liu et al., 2019a] Liu, D., Zhang, H., Wu, F., and Zha, Z.-J. (2019a). Learning to Assemble Neural Module Tree Networks for Visual Grounding. In *ICCV*.

[Liu et al., 2021a] Liu, F., Bugliarello, E., Ponti, E. M., Reddy, S., Collier, N., and Elliott, D. (2021a). Visually grounded reasoning across languages and cultures. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 10467–10485, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

[Liu et al., 2020] Liu, W., Zhou, P., Zhao, Z., Wang, Z., Ju, Q., Deng, H., and Wang, P. (2020). K-BERT: Enabling language representation with knowledge graph. In *Proceedings of AAAI 2020*.

[Liu et al., 2019b] Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., and Stoyanov, V. (2019b). Roberta: A robustly optimized bert pretraining approach.

[Liu et al., 2021b] Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., and Guo, B. (2021b). Swin transformer: Hierarchical vision transformer using shifted

windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10012–10022.

[Liu et al., 2022] Liu, Z., Ning, J., Cao, Y., Wei, Y., Zhang, Z., Lin, S., and Hu, H. (2022). Video swin transformer. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3202–3211.

[Long et al., 2021] Long, Q., Wang, M., and Li, L. (2021). Generative imagination elevates machine translation. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5738–5748, Online. Association for Computational Linguistics.

[Loshchilov and Hutter, 2018] Loshchilov, I. and Hutter, F. (2018). Decoupled weight decay regularization. In *International Conference on Learning Representations*.

[Lu et al., 2019] Lu, J., Batra, D., Parikh, D., and Lee, S. (2019). Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. *Advances in neural information processing systems*, 32.

[Lu et al., 2023] Lu, P., Qiu, L., Yu, W., Welleck, S., and Chang, K.-W. (2023). A survey of deep learning for mathematical reasoning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14605–14631, Toronto, Canada. Association for Computational Linguistics.

[Lu et al., 2022] Lu, Y., Zhu, W., Wang, X., Eckstein, M., and Wang, W. Y. (2022). Imagination-Augmented Natural Language Understanding. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4392–4402, Seattle, United States. Association for Computational Linguistics.

[Luo et al., 2020a] Luo, G., Zhou, Y., Ji, R., Sun, X., Su, J., Lin, C.-W., and Tian, Q. (2020a). Cascade grouped attention network for referring expression segmentation. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 1274–1282.

[Luo et al., 2020b] Luo, G., Zhou, Y., Sun, X., Cao, L., Wu, C., Deng, C., and Ji, R. (2020b). Multi-task collaborative network for joint referring expression comprehension and segmentation. In *CVPR*, pages 10034–10043.

[Luo et al., 2020c] Luo, H., Ji, L., Shi, B., Huang, H., Duan, N., Li, T., Li, J., Bharti, T., and Zhou, M. (2020c). Univl: A unified video and language pre-training model for multimodal understanding and generation. *arXiv preprint arXiv:2002.06353*.

[Luo et al., 2022] Luo, H., Ji, L., Zhong, M., Chen, Y., Lei, W., Duan, N., and Li, T. (2022). Clip4clip: An empirical study of clip for end to end video clip retrieval and captioning. *Neurocomputing*, 508:293–304.

[Lupyan and Clark, 2015] Lupyan, G. and Clark, A. (2015). Words and the world: Predictive coding and the language-perception-cognition interface. *Current Directions in Psychological Science*, 24(4):279–284. Publisher: Sage Publications Sage CA: Los Angeles, CA.

[Lv et al., 2020] Lv, S., Guo, D., Xu, J., Tang, D., Duan, N., Gong, M., Shou, L., Jiang, D., Hu, G. C., and Songlin (2020). Graph-based reasoning over heterogeneous external knowledge for commonsense question answering. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, New York, USA*, pages 8449–8456. AAAI Press.

[Ma et al., 2022] Ma, Y., Xu, G., Sun, X., Yan, M., Zhang, J., and Ji, R. (2022). X-clip: End-to-end multi-grained contrastive learning for video-text retrieval. In *Proceedings of the 30th ACM International Conference on Multimedia*, page 638–647.

[Maas et al., 2013] Maas, A. L., Hannun, A. Y., and Ng, A. Y. (2013). Rectifier nonlinearities improve neural network acoustic models. In *ICML*, volume 30, page 3.

[Malinowski and Fritz, 2014] Malinowski, M. and Fritz, M. (2014). A multi-world approach to question answering about real-world scenes based on uncertain input. In *NeurIPS*, pages 1682–1690.

[Manjunatha et al., 2018] Manjunatha, V., Iyyer, M., Boyd-Graber, J., and Davis, L. (2018). Learning to color from language. In *NAACL-HLT, vol. 2*, pages 764–769.

[Mao et al., 2016] Mao, J., Huang, J., Toshev, A., Camburu, O., Yuille, A., and Murphy, K. (2016). Generation and Comprehension of Unambiguous Object Descriptions. *CVPR*.

[Marcus et al., 1993] Marcus, M. P., Marcinkiewicz, M. A., and Santorini, B. (1993). Building a large annotated corpus of english: the penn treebank. *Computational Linguistics*, 19(2):313–330.

[Margffoy-Tuay et al., 2018] Margffoy-Tuay, E., Pérez, J. C., Botero, E., and Arbeláez, P. (2018). Dynamic multimodal instance segmentation guided by natural language queries. In *ECCV*, pages 630–645.

[Miech et al., 2020] Miech, A., Alayrac, J.-B., Laptev, I., Sivic, J., and Zisserman, A. (2020). Rareact: A video dataset of unusual interactions. *arXiv preprint arXiv:2008.01018*.

[Miech et al., 2019] Miech, A., Zhukov, D., Alayrac, J.-B., Tapaswi, M., Laptev, I., and Sivic, J. (2019). Howto100m: Learning a text-video embedding by watching hundred million narrated video clips. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.

[Misra et al., 2018] Misra, D., Bennett, A., Blukis, V., Niklasson, E., Shatkhin, M., and Artzi, Y. (2018). Mapping instructions to actions in 3D environments with visual goal prediction. In *EMNLP*, pages 2667–2678.

[Momeni et al., 2023] Momeni, L., Caron, M., Nagrani, A., Zisserman, A., and Schmid, C. (2023). Verbs in action: Improving verb understanding in video-language models.

[Moor et al., 2023] Moor, M., Huang, Q., Wu, S., Yasunaga, M., Zakka, C., Dalmia, Y., Reis, E. P., Rajpurkar, P., and Leskovec, J. (2023). Med-flamingo: A multimodal medical few-shot learner. arXiv:2307.15189.

[Mottaghi et al., 2016] Mottaghi, R., Bagherinezhad, H., Rastegari, M., and Farhadi, A. (2016). Newtonian scene understanding: Unfolding the dynamics of objects in static images. In *CVPR*, pages 3521–3529.

[Nagaraja et al., 2016] Nagaraja, V. K., Morariu, V. I., and Davis, L. S. (2016). Modeling Context Between Objects for Referring Expression Understanding. *Lecture Notes in Computer Science*, pages 792–807.

[Nooralahzadeh and Sennrich, 2023] Nooralahzadeh, F. and Sennrich, R. (2023). Improving the cross-lingual generalisation in visual question answering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 13419–13427.

[Ordonez et al., 2011] Ordonez, V., Kulkarni, G., and Berg, T. L. (2011). Im2text: Describing images using 1 million captioned photographs. In *Neural Information Processing Systems (NIPS)*.

[Parcalabescu et al., 2022] Parcalabescu, L., Cafagna, M., Muradjan, L., Frank, A., Calixto, I., and Gatt, A. (2022). VALSE: A task-independent benchmark for vision and language models centered on linguistic phenomena. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1:*

Long Papers), pages 8253–8280, Dublin, Ireland. Association for Computational Linguistics.

[Parcalabescu et al., 2021] Parcalabescu, L., Gatt, A., Frank, A., and Calixto, I. (2021). Seeing past words: Testing the cross-modal capabilities of pretrained V&L models on counting tasks. In *Proceedings of the 1st Workshop on Multimodal Semantic Representations (MMSR)*, pages 32–44, Groningen, Netherlands (Online). Association for Computational Linguistics.

[Park et al., 2022] Park, J. S., Shen, S., Farhadi, A., Darrell, T., Choi, Y., and Rohrbach, A. (2022). Exposing the limits of video-text models through contrast sets. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3574–3586, Seattle, United States. Association for Computational Linguistics.

[Pennington et al., 2014a] Pennington, J., Socher, R., and Manning, C. D. (2014a). Glove: Global vectors for word representation. In *EMNLP*, pages 1532–1543.

[Pennington et al., 2014b] Pennington, J., Socher, R., and Manning, C. D. (2014b). Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

[Perez et al., 2018] Perez, E., Strub, F., Vries, H. d., Dumoulin, V., and Courville, A. C. (2018). FiLM: Visual Reasoning with a General Conditioning Layer. In *AAAI*.

[Petersen et al., 2023] Petersen, F., Schubotz, M., Greiner-Petter, A., and Gipp, B. (2023). Neural machine translation for mathematical formulae. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11534–11550, Toronto, Canada. Association for Computational Linguistics.

[Pikuliak et al., 2022] Pikuliak, M., Grivalský, Š., Konôpka, M., Blšták, M., Tamajka, M., Bachratý, V., Simko, M., Balážik, P., Trnka, M., and Uhlárik, F. (2022). SlovakBERT: Slovak masked language model. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 7156–7168, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

[Plummer et al., 2015] Plummer, B. A., Wang, L., Cervantes, C. M., Caicedo, J. C., Hockenmaier, J., and Lazebnik, S. (2015). Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In *Proceedings of the IEEE international conference on computer vision*, pages 2641–2649.

[Radford et al., 2021] Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al. (2021). Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR.

[Radford et al., 2019] Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I., et al. (2019). Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

[Raffel et al., 2020] Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., Liu, P. J., et al. (2020). Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21(140):1–67.

[Ramesh et al., 2021] Ramesh, A., Pavlov, M., Goh, G., Gray, S., Voss, C., Radford, A., Chen, M., and Sutskever, I. (2021). Zero-shot text-to-image generation. In Meila, M. and Zhang, T., editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 8821–8831. PMLR.

[Ranftl et al., 2021] Ranftl, R., Bochkovskiy, A., and Koltun, V. (2021). Vision

transformers for dense prediction. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 12179–12188.

[Ren et al., 2015] Ren, M., Kiros, R., and Zemel, R. (2015). Exploring models and data for image question answering. In *NeurIPS*, pages 2953–2961.

[Ren et al., 2017] Ren, S., He, K., Girshick, R., and Sun, J. (2017). Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *TPAMI*, 39(6):1137–1149.

[Ronneberger et al., 2015] Ronneberger, O., Fischer, P., and Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*, pages 234–241.

[Rosenberg et al., 2021] Rosenberg, D., Gat, I., Feder, A., and Reichart, R. (2021). Are VQA systems RAD? Measuring robustness to augmented data with focused interventions. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 61–70, Online. Association for Computational Linguistics.

[Rumelhart and McClelland, 1987] Rumelhart, D. E. and McClelland, J. L. (1987). Learning internal representations by error propagation.

[Runia et al., 2018] Runia, T. F. H., Snoek, C. G. M., and Smeulders, A. W. M. (2018). Real-world repetition estimation by div, grad and curl. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[Safaya et al., 2022] Safaya, A., Kurtuluş, E., Goktogan, A., and Yuret, D. (2022). Mukayese: Turkish NLP strikes back. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 846–863, Dublin, Ireland. Association for Computational Linguistics.

[Sampat et al., 2021] Sampat, S. K., Kumar, A., Yang, Y., and Baral, C. (2021). CLEVR_HYP: A challenge dataset and baselines for visual question answering with hypothetical actions over images. In *NAACL-HLT*.

[Sanh et al., 2019] Sanh, V., Debut, L., Chaumond, J., and Wolf, T. (2019). Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *ArXiv*, abs/1910.01108.

[Sanh et al., 2021] Sanh, V., Webson, A., Raffel, C., Bach, S., Sutawika, L., Alyafeai, Z., Chaffin, A., Stiegler, A., Raja, A., Dey, M., et al. (2021). Multitask prompted training enables zero-shot task generalization. In *International Conference on Learning Representations*.

[Sap et al., 2019] Sap, M., Rashkin, H., Chen, D., Le Bras, R., and Choi, Y. (2019). Social IQa: Commonsense reasoning about social interactions. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4463–4473, Hong Kong, China. Association for Computational Linguistics.

[Sap et al., 2020] Sap, M., Shwartz, V., Bosselut, A., Choi, Y., and Roth, D. (2020). Commonsense reasoning for natural language processing. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: Tutorial Abstracts*, pages 27–33, Online. Association for Computational Linguistics.

[See et al., 2017] See, A., Liu, P. J., and Manning, C. D. (2017). Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1073–1083, Vancouver, Canada. Association for Computational Linguistics.

[Seo et al., 2022] Seo, P. H., Nagrani, A., Arnab, A., and Schmid, C. (2022). End-to-end generative pretraining for multimodal video captioning. In *Proceedings of*

the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 17959–17968.

[Sharma et al., 2018] Sharma, P., Ding, N., Goodman, S., and Soricut, R. (2018). Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2556–2565, Melbourne, Australia. Association for Computational Linguistics.

[Shekhar et al., 2017a] Shekhar, R., Pezzelle, S., Herbelot, A., Nabi, M., Sangineto, E., and Bernardi, R. (2017a). Vision and language integration: Moving beyond objects. In *IWCS 2017 — 12th International Conference on Computational Semantics — Short papers*.

[Shekhar et al., 2017b] Shekhar, R., Pezzelle, S., Klimovich, Y., Herbelot, A., Nabi, M., Sangineto, E., and Bernardi, R. (2017b). FOIL it! find one mismatch between image and language caption. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 255–265, Vancouver, Canada. Association for Computational Linguistics.

[Shen et al., 2020] Shen, T., Mao, Y., He, P., Long, G., Trischler, A., and Chen, W. (2020). Exploiting structured knowledge in text via graph-guided representation learning. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8980–8994, Online. Association for Computational Linguistics.

[Shi et al., 2018] Shi, H., Li, H., Meng, F., and Wu, Q. (2018). Key-Word-Aware Network for Referring Expression Image Segmentation. In *ECCV*.

[Shi et al., 2015] Shi, X., Chen, Z., Wang, H., Yeung, D.-Y., Wong, W.-k., and WOO, W.-c. (2015). Convolutional LSTM Network: A Machine Learning Approach for Precipitation Nowcasting. In *NeurIPS*, pages 802–810.

[Simonyan and Zisserman, 2014] Simonyan, K. and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

[Singh et al., 2022] Singh, G., Wu, Y.-F., and Ahn, S. (2022). Simple unsupervised object-centric learning for complex and naturalistic videos. *Advances in Neural Information Processing Systems*, 35:18181–18196.

[Singhal et al., 2023] Singhal, K., Tu, T., Gottweis, J., Sayres, R., Wulczyn, E., Hou, L., Clark, K., Pfohl, S., Cole-Lewis, H., Neal, D., Schaekermann, M., Wang, A., Amin, M., Lachgar, S., Mansfield, P., Prakash, S., Green, B., Dominowska, E., y Arcas, B. A., Tomasev, N., Liu, Y., Wong, R., Semturs, C., Mahdavi, S. S., Barral, J., Webster, D., Corrado, G. S., Matias, Y., Azizi, S., Karthikesalingam, A., and Natarajan, V. (2023). Towards expert-level medical question answering with large language models.

[Soomro et al., 2012] Soomro, K., Zamir, A. R., and Shah, M. (2012). Ucf101: A dataset of 101 human actions classes from videos in the wild.

[Srivastava et al., 2014] Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. *JMLR*, 15(1):1929–1958.

[Suhr et al., 2019] Suhr, A., Zhou, S., Zhang, A., Zhang, I., Bai, H., and Artzi, Y. (2019). A corpus for reasoning about natural language grounded in photographs. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6418–6428, Florence, Italy. Association for Computational Linguistics.

[Sun et al., 2023] Sun, X., Chen, P., Chen, L., Li, C., Li, T. H., Tan, M., and Gan, C. (2023). Masked motion encoding for self-supervised video representation learning. In *The IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

[Sutskever et al., 2014] Sutskever, I., Vinyals, O., and Le, Q. V. (2014). Sequence to sequence learning with neural networks. *Advances in neural information processing systems*, 27.

[Tan and Bansal, 2019a] Tan, H. and Bansal, M. (2019a). LXMERT: Learning cross-modality encoder representations from transformers. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5100–5111, Hong Kong, China. Association for Computational Linguistics.

[Tan and Bansal, 2019b] Tan, H. and Bansal, M. (2019b). Lxmert: Learning cross-modality encoder representations from transformers. In *EMNLP-IJCNLP*, pages 5100–5111.

[Tan and Bansal, 2020] Tan, H. and Bansal, M. (2020). Vokenization: Improving language understanding with contextualized, visual-grounded supervision. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2066–2080, Online. Association for Computational Linguistics.

[Tapaswi et al., 2016] Tapaswi, M., Zhu, Y., Stiefelhagen, R., Torralba, A., Urtasun, R., and Fidler, S. (2016). MovieQA: Understanding stories in movies through question-answering. In *CVPR*, pages 4631–4640.

[Thomason et al., 2016] Thomason, J., Sinapov, J., Svetlik, M., Stone, P., and Mooney, R. J. (2016). Learning multi-modal grounded linguistic semantics by playing" i spy". In *IJCAI*, pages 3477–3483.

[Thrush et al., 2022] Thrush, T., Jiang, R., Bartolo, M., Singh, A., Williams, A., Kiela, D., and Ross, C. (2022). Winoground: Probing vision and language models for visio-linguistic compositionality. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5238–5248.

[Tran et al., 2018] Tran, D., Wang, H., Torresani, L., Ray, J., LeCun, Y., and Paluri, M. (2018). A closer look at spatiotemporal convolutions for action recognition. In *CVPR*, pages 6450–6459.

[Unal et al., 2016] Unal, M. E., Citamak, B., Yagcioglu, S., Erdem, A., Erdem, E., Cinbis, N. I., and Cakici, R. (2016). Tasviret: A benchmark dataset for automatic turkish description generation from images. In *2016 24th signal processing and communication application conference (SIU)*, pages 1977–1980. IEEE.

[Vaswani et al., 2017] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, \., and Polosukhin, I. (2017). Attention is all you need. In *NeurIPS*, pages 5998–6008.

[Verma et al., 2023] Verma, Y., Jangra, A., Verma, R., and Saha, S. (2023). Large scale multi-lingual multi-modal summarization dataset. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 3620–3632, Dubrovnik, Croatia. Association for Computational Linguistics.

[Vinyals et al., 2015] Vinyals, O., Toshev, A., Bengio, S., and Erhan, D. (2015). Show and tell: A neural image caption generator. *CVPR*.

[Wagner et al., 2018] Wagner, M., Basevi, H., Shetty, R., Li, W., Malinowski, M., Fritz, M., and Leonardis, A. (2018). Answering visual what-if questions: From actions to predicted scene descriptions. In *ECCV Workshops*.

[Wang et al., 2019a] Wang, A., Pruksachatkun, Y., Nangia, N., Singh, A., Michael, J., Hill, F., Levy, O., and Bowman, S. R. (2019a). SuperGLUE: A stickier benchmark for general-purpose language understanding systems. *arXiv preprint 1905.00537*.

[Wang et al., 2018] Wang, A., Singh, A., Michael, J., Hill, F., Levy, O., and Bowman,

S. R. (2018). Glue: A multi-task benchmark and analysis platform for natural language understanding. In *International Conference on Learning Representations*.

[Wang et al., 2019b] Wang, A., Singh, A., Michael, J., Hill, F., Levy, O., and Bowman, S. R. (2019b). GLUE: A multi-task benchmark and analysis platform for natural language understanding. In the Proceedings of ICLR.

[Wang et al., 2022a] Wang, A. J., Ge, Y., Yan, R., Ge, Y., Lin, X., Cai, G., Wu, J., Shan, Y., Qie, X., and Shou, M. Z. (2022a). All in one: Exploring unified video-language pre-training. *arXiv preprint arXiv:2203.07303*.

[Wang et al., 2022b] Wang, J., Chen, D., Wu, Z., Luo, C., Zhou, L., Zhao, Y., Xie, Y., Liu, C., Jiang, Y.-G., and Yuan, L. (2022b). OmniVL: One foundation model for image-language and video-language tasks. In Oh, A. H., Agarwal, A., Belgrave, D., and Cho, K., editors, *Advances in Neural Information Processing Systems*.

[Wang et al., 2019c] Wang, P., Wu, Q., Cao, J., Shen, C., Gao, L., and Hengel, A. v. d. (2019c). Neighbourhood Watch: Referring Expression Comprehension via Language-Guided Graph Attention Networks. *CVPR*.

[Wang et al., 2023a] Wang, R., Chen, D., Wu, Z., Chen, Y., Dai, X., Liu, M., Yuan, L., and Jiang, Y.-G. (2023a). Masked video distillation: Rethinking masked feature modeling for self-supervised video representation learning. In *CVPR*.

[Wang et al., 2023b] Wang, T., Lin, K., Li, L., Lin, C.-C., Yang, Z., Zhang, H., Liu, Z., and Wang, L. (2023b). Equivariant similarity for vision-language foundation models. *arXiv preprint arXiv:2303.14465*.

[Wang et al., 2019d] Wang, X., Wu, J., Chen, J., Li, L., Wang, Y.-F., and Wang, W. Y. (2019d). Vatex: A large-scale, high-quality multilingual dataset for video-and-language research. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4581–4591.

[Wang et al., 2021] Wang, X., Zhu, L., and Yang, Y. (2021). T2vlad: global-local sequence alignment for text-video retrieval. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5079–5088.

[Wang et al., 2022c] Wang, Z., Wu, Z., Agarwal, D., and Sun, J. (2022c). MedCLIP: Contrastive learning from unpaired medical images and text. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 3876–3887, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

[Wei et al., 2022] Wei, J., Wang, X., Schuurmans, D., Bosma, M., Xia, F., Chi, E., Le, Q. V., Zhou, D., et al. (2022). Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35:24824–24837.

[Winograd, 1972] Winograd, T. (1972). Understanding natural language. *Cognitive psychology*, 3(1):1–191.

[Wolf et al., 2019] Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., et al. (2019). Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*.

[Wolff and Barbey, 2015] Wolff, P. and Barbey, A. K. (2015). Causal reasoning with forces. *Front. Hum. Neurosci.*, 9:1.

[Wu et al., 2021] Wu, B., Yu, S., Chen, Z., Tenenbaum, J. B., and Gan, C. (2021). STAR: A benchmark for situated reasoning in real-world videos. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*.

[Wu et al., 2023] Wu, C., Zhang, X., Zhang, Y., Wang, Y., and Xie, W. (2023).

Medklip: Medical knowledge enhanced language-image pre-training. *Proceedings of the IEEE/CVF International Conference on Computer Vision*.

[Xia et al., 2022] Xia, F., Li, B., Weng, Y., He, S., Liu, K., Sun, B., Li, S., and Zhao, J. (2022). MedConQA: Medical conversational question answering system based on knowledge graphs. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 148–158, Abu Dhabi, UAE. Association for Computational Linguistics.

[Xiao et al., 2021] Xiao, J., Shang, X., Yao, A., and Chua, T.-S. (2021). Next-qa: Next phase of question-answering to explaining temporal actions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9777–9786.

[Xie et al., 2018] Xie, S., Sun, C., Huang, J., Tu, Z., and Murphy, K. (2018). Rethinking spatiotemporal feature learning: Speed-accuracy trade-offs in video classification. In *Proceedings of the European conference on computer vision (ECCV)*, pages 305–321.

[Xiong et al., 2019] Xiong, W., Du, J., Wang, W. Y., and Stoyanov, V. (2019). Pretrained encyclopedia: Weakly supervised knowledge-pretrained language model. In *International Conference on Learning Representations*.

[Xu et al., 2017] Xu, D., Zhao, Z., Xiao, J., Wu, F., Zhang, H., He, X., and Zhuang, Y. (2017). Video question answering via gradually refined attention over appearance and motion. In *Proceedings of the 25th ACM international conference on Multimedia*, pages 1645–1653.

[Xu et al., 2021a] Xu, H., Ghosh, G., Huang, P.-Y., Okhonko, D., Aghajanyan, A., Metze, F., Zettlemoyer, L., and Feichtenhofer, C. (2021a). VideoCLIP: Contrastive pre-training for zero-shot video-text understanding. In *Proceedings of the 2021*

Conference on Empirical Methods in Natural Language Processing, pages 6787–6800, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

[Xu et al., 2016] Xu, J., Mei, T., Yao, T., and Rui, Y. (2016). Msr-vtt: A large video description dataset for bridging video and language. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5288–5296.

[Xu et al., 2015] Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A., Salakhudinov, R., Zemel, R., and Bengio, Y. (2015). Show, attend and tell: Neural image caption generation with visual attention. In *ICML*, pages 2048–2057.

[Xu et al., 2021b] Xu, Y., Zhu, C., Xu, R., Liu, Y., Zeng, M., and Huang, X. (2021b). Fusing context into knowledge graph for commonsense question answering. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 1201–1207, Online. Association for Computational Linguistics.

[Xue et al., 2021] Xue, L., Constant, N., Roberts, A., Kale, M., Al-Rfou, R., Siddhant, A., Barua, A., and Raffel, C. (2021). mT5: A massively multilingual pre-trained text-to-text transformer. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 483–498, Online. Association for Computational Linguistics.

[Yang et al., 2022] Yang, A., Miech, A., Sivic, J., Laptev, I., and Schmid, C. (2022). Zero-shot video question answering via frozen bidirectional language models. In *Advances in Neural Information Processing Systems*.

[Yang et al., 2020a] Yang, S., Li, G., and Yu, Y. (2020a). Graph-structured referring expressions reasoning in the wild.

[Yang et al., 2020b] Yang, S., Li, G., and Yu, Y. (2020b). Propagating over phrase relations for one-stage visual grounding. In *ECCV*, pages 589–605. Springer.

[Yang et al., 2021] Yang, S., Xia, M., Li, G., Zhou, H.-Y., and Yu, Y. (2021). Bottom-up shift and reasoning for referring image segmentation. In *CVPR*, pages 11266–11275.

[Yang et al., 2020c] Yang, Z., Chen, T., Wang, L., and Luo, J. (2020c). Improving one-stage visual grounding by recursive sub-query construction. In *ECCV*, pages 387–404. Springer.

[Yang et al., 2019] Yang, Z., Gong, B., Wang, L., Huang, W., Yu, D., and Luo, J. (2019). A fast and accurate one-stage approach to visual grounding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4683–4693.

[Ye et al., 2020] Ye, L., Liu, Z., and Wang, Y. (2020). Dual convolutional lstm network for referring image segmentation. *TMM*, 22(12):3224–3235.

[Ye et al., 2019] Ye, L., Rochan, M., Liu, Z., and Wang, Y. (2019). Cross-Modal Self-Attention Network for Referring Image Segmentation. *CVPR*.

[Yi et al., 2020] Yi, K., Gan, C., Li, Y., Kohli, P., Wu, J., Torralba, A., and Tenenbaum, J. B. (2020). Clevrer: Collision events for video representation and reasoning. In *ICLR*.

[Yu et al., 2022a] Yu, D., Zhu, C., Yang, Y., and Zeng, M. (2022a). Jaket: Joint pre-training of knowledge graph and language understanding. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(10):11630–11638.

[Yu et al., 2016a] Yu, H., Wang, J., Huang, Z., Yang, Y., and Xu, W. (2016a). Video paragraph captioning using hierarchical recurrent neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4584–4593.

[Yu et al., 2018] Yu, L., Lin, Z., Shen, X., Yang, J., Lu, X., Bansal, M., and Berg, T. L. (2018). MAttNet: Modular Attention Network for Referring Expression Comprehension. *CVPR*.

[Yu et al., 2016b] Yu, L., Poirson, P., Yang, S., Berg, A. C., and Berg, T. L. (2016b). Modeling Context in Referring Expressions. *Lecture Notes in Computer Science*, pages 69–85.

[Yu et al., 2022b] Yu, W., Zhu, C., Fang, Y., Yu, D., Wang, S., Xu, Y., Zeng, M., and Jiang, M. (2022b). Dict-BERT: Enhancing language model pre-training with dictionary. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 1907–1918, Dublin, Ireland. Association for Computational Linguistics.

[Yu et al., 2019] Yu, Z., Xu, D., Yu, J., Yu, T., Zhao, Z., Zhuang, Y., and Tao, D. (2019). Activitynet-qa: A dataset for understanding complex web videos via question answering. In *AAAI*, volume 33, pages 9127–9134.

[Zellers et al., 2021a] Zellers, R., Holtzman, A., Peters, M., Mottaghi, R., Kembhavi, A., Farhadi, A., and Choi, Y. (2021a). PIGLeT: Language grounding through neuro-symbolic interaction in a 3D world. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 2040–2050, Online. Association for Computational Linguistics.

[Zellers et al., 2022] Zellers, R., Lu, J., Lu, X., Yu, Y., Zhao, Y., Salehi, M., Kusupati, A., Hessel, J., Farhadi, A., and Choi, Y. (2022). Merlot reserve: Multimodal neural script knowledge through vision and language and sound. In *CVPR*.

[Zellers et al., 2021b] Zellers, R., Lu, X., Hessel, J., Yu, Y., Park, J. S., Cao, J., Farhadi, A., and Choi, Y. (2021b). Merlot: Multimodal neural script knowledge models. *Advances in Neural Information Processing Systems*, 34:23634–23651.

[Zhang et al., 2016a] Zhang, P., Goyal, Y., Summers-Stay, D., Batra, D., and Parikh, D. (2016a). Yin and yang: Balancing and answering binary visual questions. In *CVPR*, pages 5014–5022.

[Zhang et al., 2016b] Zhang, R., Isola, P., and Efros, A. A. (2016b). Colorful image colorization. In *ECCV*, pages 649–666.

[Zhang et al., 2018] Zhang, R., Isola, P., Efros, A. A., Shechtman, E., and Wang, O. (2018). The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*.

[Zhang et al., 2022] Zhang, S., Roller, S., Goyal, N., Artetxe, M., Chen, M., Chen, S., Dewan, C., Diab, M., Li, X., Lin, X. V., et al. (2022). Opt: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068*.

[Zhang et al., 2019] Zhang, Z., Han, X., Liu, Z., Jiang, X., Sun, M., and Liu, Q. (2019). ERNIE: Enhanced language representation with informative entities. In *Proceedings of ACL 2019*.

[Zhang et al., 2023] Zhang, Z., Hu, X., Zhang, J., Zhang, Y., Wang, H., Qu, L., and Xu, Z. (2023). FEDLEGAL: The first real-world federated learning benchmark for legal NLP. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3492–3507, Toronto, Canada. Association for Computational Linguistics.

[Zhou et al., 2021] Zhou, B., Richardson, K., Ning, Q., Khot, T., Sabharwal, A., and Roth, D. (2021). Temporal reasoning on implicit events from distant supervision. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1361–1371, Online. Association for Computational Linguistics.

[Zhu et al., 2022a] Zhu, C., Xu, Y., Ren, X., Lin, B. Y., Jiang, M., and Yu, W. (2022a). Knowledge-augmented methods for natural language processing. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: Tutorial Abstracts*, pages 12–20, Dublin, Ireland. Association for Computational Linguistics.

[Zhu and Yang, 2020] Zhu, L. and Yang, Y. (2020). Actbert: Learning global-local video-text representations. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8746–8755.

[Zhu and Bhat, 2020] Zhu, W. and Bhat, S. (2020). GRUEN for evaluating linguistic quality of generated text. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 94–108, Online. Association for Computational Linguistics.

[Zhu and Bhat, 2021] Zhu, W. and Bhat, S. (2021). Euphemistic phrase detection by masked language model. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 163–168, Punta Cana, Dominican Republic. Association for Computational Linguistics.

[Zhu et al., 2021] Zhu, W., Gong, H., Bansal, R., Weinberg, Z., Christin, N., Fanti, G., and Bhat, S. (2021). Self-supervised euphemism detection and identification for content moderation. In *42nd IEEE Symposium on Security and Privacy*.

[Zhu et al., 2022b] Zhu, X., Zhu, J., Li, H., Wu, X., Li, H., Wang, X., and Dai, J. (2022b). Uni-perceiver: Pre-training unified architecture for generic perception for zero-shot and few-shot tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16804–16815.

[Zhu et al., 2016a] Zhu, Y., Groth, O., Bernstein, M., and Fei-Fei, L. (2016a). Visual7w: Grounded question answering in images. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4995–5004.

[Zhu et al., 2016b] Zhu, Y., Groth, O., Bernstein, M., and Fei-Fei, L. (2016b). Visual7w: Grounded question answering in images. In *CVPR*, pages 4995–5004.

[Zou et al., 2019] Zou, C., Mo, H., Gao, C., Du, R., and Fu, H. (2019). Language-based colorization of scene sketches. *TOG)*, 38(6):1–16. Publisher: ACM New York, NY, USA.