Visual Understanding through Natural Language

by

Lisa Anne Marie Hendricks

A dissertation submitted in partial satisfaction of the

requirements for the degree of

Doctor of Philosophy

in

Engineering - Electrical Engineering and Computer Science

in the

Graduate Division

of the

University of California, Berkeley

Committee in charge:

Professor Trevor Darrell, Chair
Professor Jitendra Malik
Assistant Professor David Bamman

Spring 2019

Visual Understanding through Natural Language

Abstract

Visual Understanding through Natural Language

by

Lisa Anne Marie Hendricks

Doctor of Philosophy in Engineering - Electrical Engineering and Computer Science

University of California, Berkeley

Professor Trevor Darrell, Chair

Powered by deep convolutional networks and large scale visual datasets, modern computer vision systems are capable of accurately recognizing thousands of visual categories. However, images contain much more than categorical labels: they contain information about where objects are located (in a forest or in a kitchen?), what attributes an object has (red or blue?), and how objects interact with other objects in a scene (is the child sitting on a sofa, or running in a field?). Natural language provides an efficient and intuitive way for visual systems to convey important information about a visual scene.

We begin by considering a fundamental task as the intersection of language and vision: image captioning, in which a system receives an image as input and outputs a natural language sentence that describes the image. We consider two important shortcomings in modern image captioning models. First, in order to describe an object, like "otter", captioning models require pairs of sentences and images which include the object "otter". In Chapter 2, we build models that can learn an object like "otter" from classification data, which is abundant and easy to collect, then compose novel sentences at test time describing "otter", without any "otter" image caption examples at train time. Second, visual description models can be heavily driven by biases found in the training dataset. This can lead to object hallucination in which models hallucinate objects not present in an image. In Chapter 3, we propose tools to analyze language bias through the lens of object hallucination. Language bias can also lead to bias amplification; e.g., if otters occur in 70% of train images, at test time a model might predict that otters occur in 85% of test images. We propose the Equalizer model in Chapter 4 to mitigate such bias in a special, yet important, case: gender bias.

Moving on from captioning, we consider how systems which provide natural language text about an image can be used to help humans better understand an AI system. In Chapter 5, we propose to generate visual explanations with natural language, which rationalize the output of a deep visual classifier. We show these explanations can help humans understand when to accept or reject decisions made by an AI agent. Finally, in Chapter 6, we consider a new task at the intersection of language and vision: moment localization in videos with natural language. We detail the collection of a large scale dataset for this task, as well as the first models for moment localization.

To my sisters, Dianne and Khalida

Contents

Acknowledgments

First and foremost, I would like to thank my advisor Trevor Darrell. I did not apply to Berkeley as a computer vision student, but I remember my first conversation with Trevor during visit days – discussing how the reflection of a glass made it more difficult to recognize for a computer. With his guidance, I have been able to discover a research field I am truly passionate about. Thank you for your patience, support, and motivation through my studies. Additionally, I would like to thank Marti Hearst and Alexei Efros for serving on my qualification committee, David Bamman for serving on my thesis committee, and Jitendra Malik for serving on both my qualification and thesis committees.

I would also like to thank all those who have mentored me through my PhD, in particular Marcus Rohrbach, Bryan Russell, and Zeynep Akata. Marcus was the first post-doc I worked closely with at UC Berkeley and through his close guidance I was able to transform a hard, undefined question into my first paper. Bryan was my mentor during two wonderful internships at Adobe. Through his support and guidance I pursued a project which was different from what I would have pursued at Berkeley and which makes up the penultimate chapter of this thesis. I grew so much as a researcher from our work together. Zeynep Akata was a visiting researcher at Berkeley and has helped me shape my thoughts on explainable AI. In addition to having many papers together, Zeynep is an excellent example of how to be a confident researcher. Whenever I am in a situation where I know I need to be confident in myself, I think “What would Zeynep do?” I would also like to thank Ray Mooney, Kate Saenko, Bernt Schiele, my Adobe internship collaborators Eli Shechtman, Oliver Wang, and Josef Sivic, and my Facebook internship hosts Devi Parikh and Dhruv Batra for providing additional mentorship during my PhD.

Throughout my years at Berkeley, I have been fortunate to have so many collaborators, both at Berkeley and elsewhere: Jeff Donahue, Subhashini Venugopalan, Sergio Guadarrama, Dong Huk Park, Ronghang Hu, Kaylee Burns, and Anna Rohrbach. In addition, though I did not collaborate with everyone in the vision groups at UC Berkeley, I would like to thank them all for making Berkeley such an intellectually stimulating environment. In particular, I would like to thank Judy Hoffman and Georgia Gkioxari who introduced me to Trevor during visit days after I shared some of my interests with them. Without you, my life would have been very different!

A PhD is a long journey, and I would like to thank my friends for providing emotional support, laughter, and encouragement: Benjamin Allardet, Carolyn Branecky, Hilary Purrington, and Samantha Masaki. I would like to give a special thanks to Emily Mimovich for always cheering me on, making me laugh, and sharing a life-changing trek in Nepal with me. Jeff – thanks for not only being one of my first research collaborators, but a constant pillar of support and my best friend. Finally, thank you to Celeste Riepe - who has now lived under the same roof as me for close to ten years. Not only have you been a willing subject for many of my experiments (posing for my first computational photography class in undergrad and using your bird expertise to evaluate my models), but you have been a source of encouragement throughout my PhD. I am so happy we were able to share our graduate school experience together; I cannot wait to see what we do after earning our degrees!

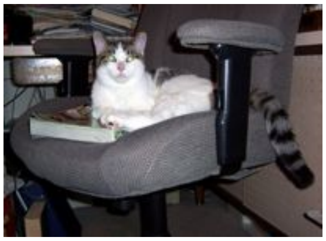
Finally, I would like to thank my family including my mother Dianne, father John, and stepmother Margaret. Thank you for providing me with everything I needed to succeed in life, and instilling the value of education in me from a young age. I would also like to thank my siblings Dianne, Khalida, James, and David. Most of all, I would like to thank my sisters Dianne and Khalida for their support and love throughout my entire life. From a young age you have taught me many things, from ice skating to helping with me science fair projects, and inspired me to reach for my dreams. Through your examples, I have learned how to navigate the world as a smart, independent, and feisty young woman. Thank you for your continuous support and love; this thesis is dedicated to you.

Chapter 1

Introduction

1.1 Motivation

Perception is key to a variety of artificial intelligence (AI) applications such as smart homes, autonomous vehicles, and accessibility tools (e.g., description tools for the visually impaired). However, for humans to effectively use and understand AI systems, it is important that they can communicate with them. Natural language is one way in which humans and AI systems can efficiently communicate about the visual world.



A black and white cat is sitting on a chair.

Figure 1.1

Jointly modeling language and vision allows AI systems to express a wide variety of information about visual scenes. For example, consider the image in Figure 1.1. Whereas a classification system might be able to name objects, a system which outputs a visual description of the scene such as “A black and white cat is sitting on a chair” not only captures the salient objects in the scene (the cat and the chair), but demonstrates recognition of important attributes (the cat is black and white) and an important relationship (the cat is sitting on a chair). Thus, from a vision perspective, jointly modeling language and vision improves visual agents by enabling them to understand and output a richer set of semantic concepts. From a language understanding perspective, jointly modeling language and vision allows for grounded language learning, in which the meaning of a word is not just determined by its linguistic context, but by what entity it refers to in the real world [Har90].

1.2 Background

Linking images and natural language has a rich history in the artificial intelligence community [Duy+02; Kul+13; JLM03; Far+10]. Early models use a pipeline approach in which individual entities are recognized (e.g., “chair”) and then mapped to sentences either via a generative process [Kul+13] or

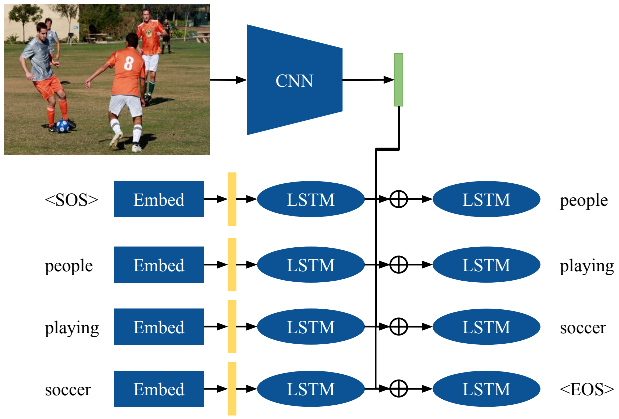


Figure 1.2: The LRCN model: one of the first end-to-end deep description models.

by retrieving sentences which include similar semantic content [Far+10]. Other work cast visual description as a machine translation task [Duy+02; Roh+13], in which an image is "translated" into a textual representation. Early work in visual description relied on hand-designed low-level features which might be task-specific. For example, to build an attribute classifier, [Far+09] combined texture descriptors [VZ05], HOG [DT05], edge detectors [Can87], and color descriptors based off pixel values for the task.

Modern image captioning systems greatly improve upon these early systems due to three important advancements. First, the seminal work of Krizhevsky et al. [KSH12] transformed the computer vision community by demonstrating that deep networks could be trained to perform difficult, large scale classification tasks much better than hand-designed features. Quickly following this discovery, others showed that features trained for a classification task on one specific dataset (e.g,. Imagenet [Den+09]) could be transferred to other datasets and tasks [Gir+14; Don+14]. Second, outside of computer vision, natural language processing also witnessed exciting results due to deep learning. In particular, sequence to sequence networks [SVL14], in which a recurrent network encoder maps an input sentence into a fixed length vector then a decoder outputs a sequence of words given the encoded vector, led to large improvements on machine translation. Finally, large scale visual description datasets were released [You+14; Che+15] which allowed for training deep models on the task of visual description. The first end-to-end deep visual description models [Don+15; Mao+15a; KFF15; Vin+15] transferred deep features extracted from classification models, built off the sequence to sequence architecture for machine translation, and relied on large scale image description datasets for training.

Contemporaneous with [Mao+15a; KFF15; Vin+15], we proposed a deep description model called the Long-Term Recurrent Convolutional Network (LRCN) model [Don+15]. As shown in Figure 1.2, the LRCN architecture generates sentences by combining deep visual features with recurrent networks. First, an image is input into a convolutional network (Figure 1.2, top) and an intermediate visual feature (in green) is extracted. To generate a sentence, a start-of-sentence token

(<SOS>) is input into a *language* LSTM. The output of this LSTM is concatenated with the visual feature and input into a second *multimodal* LSTM. The output of the second LSTM is then used to predict the next word in the sentence. At train time, usually ground truth words in a sentence are input at each time step (regardless of the predicted outputs). The prediction of each individual word in a sentence can be thought of as a classification task where the input is an image and previously generated words, and the labels consist of all the vocabulary words. Generally, a softmax cross entropy loss, the standard loss for training classification models, is optimized to train description models. At test time, predicted words are used to predict subsequent words. Other works present slight variations of LRCN. For example, [KFF15; Vin+15] used the image to initialize the hidden state of the LSTM, but did not input the image at every time step and [Mao+15a] did not have a multimodal LSTM, but a non-recurrent multimodal unit which combined linguistic and visual information.

Recent improvements in captioning include better visual representations and loss functions. Most current models include an attention mechanism [Xu+15a], in which the model focuses on a different part of the visual feature map before outputting a new word. In addition, instead of using high level global representations of images, current methods have moved toward more structured image representations. For example, [And+18a] proposes attending over extracted bounding boxes in images and [Zho+18; Lu+18] integrate a detection pipeline directly into the captioning framework. Moving towards even more structure, [Yao+18] proposed including graph convolutional networks in their description framework. Instead of relying on the cross entropy loss, improved loss functions which directly optimize for common evaluation metrics such as CIDEr [Ren+17; Liu+17], adversarial losses to encourage diverse sentences [Dai+17; She+17b], and a discriminability objective which encourages sentences to include descriptive information which can discriminate between different images in the training set [Luo+18] have been explored. The general framework of modern deep captioning models provided a backbone for progress on other other language and vision tasks as well. For example [Hu+16; Mao+16] adapted description models to natural language object localization, [Ven+15a; Ven+15b] adapted image description models to video description, and [MRF15] adapted the image description models to visual question answering (VQA). All of these tasks have evolved substantially over the last few years, and as we continue to make progress on these tasks, researchers continue to find more challenging and realistic tasks at the intersection of language and vision including visual dialog [DV+17; Das+17b], dense captioning [JKFF16], and instruction following for embodied agents [And+18b].

1.3 Thesis Goals and Contributions

Though the initial performance of end-to-end deep models on the captioning task was quite impressive, at the time models like LRCN were released, it was not clear if description models were learning to truly ground individual concepts like “cat” and “chair” in order to compose sentences, or if they were learning a mapping function between images and sentences without an understanding of the individual semantic parts. Indeed [Dev+15] showed that a nearest neighbor baseline (in which descriptions are always copied from sentences in the train set) achieved competitive performance when compared to the first captioning systems described in [Don+15; Mao+15a; KFF15; Vin+15] suggesting that models need not learn to compose novel sentences to perform well on benchmark captioning tasks. To address this issue, in the first half of this thesis we consider tasks in which models must be capable of understanding constituent visual elements which make up a visual scene (and not solely rely on priors) in order to be successful.

We start in the Chapter 2 by proposing the task of novel object captioning which requires description models to learn new visual entities (e.g., "otter") from object classification data, and then compose a novel sentence describing these objects *without* example descriptions at training time. Such models are *compositional* because they seamlessly construct sentences about new objects by combining them with linguistic expressions seen in training data. We outline an evaluation protocol for novel object captioning, present the Deep Compositional Captioner (DCC) which was the first deep description model proposed for this task, then describe the Novel Object Captioner (NOC), an improved end-to-end model for novel object captioning. Both DCC and NOC are able to describe images of objects for which no image description training examples exist.

In Chapter 3, we will take a closer look at how bias in training datasets and algorithms impact image captioning. We explore a certain kind of error image captioning models are prone to make: object hallucination, i.e., predicting an object is in an image even though it is not present. We will propose a metric, Caption Hallucination Assessment with Image Relevance (CHAIR), to measure hallucination across different models and use this to try to better understand what drives errors in captioning models.

In Chapter 4, we consider a special case of bias in image captioning: gender bias. Baseline description models amplify biases in training sets; e.g., if a "man" is present in 70% of training images, a model might predict a "man" in 90% of test images by relying solely on a training prior. We analyze gender bias in visual description models and present the Equalizer model which leads to less biased captions. In addition to more accurately describing people and their gender, our sentences are more frequently *right for the right reasons* (the model considers appropriate gender evidence when predicting the gender of a person).

The second, third, and fourth chapter of the thesis focus on how to generate better captions. Chpater 5 focuses on how to provide explanations of AI systems given a model that can produce relevant text about an image. We propose a model which, given an image and a classification decision, outputs justification text (e.g., "This is a Bronzed Cowbird because this bird is black with red eyes"). We posit that textual explanations must be *image relevant* (discuss objects and attributes present in an image) and *class discriminative* (discuss information which is actually relevant for making a decision). We first propose a new loss to encourage explanations to discuss discriminative information, then integrate visual grounding to ensure that mentioned attributes are grounded in image evidence. We demonstrate that our explanations can help humans decide when to accept a decision made by an AI agent.

In Chapter 6, we move away from text generation and consider a new task at the intersection of language and vision: moment localization with natural language. In moment localization, a text query (e.g., "the man walks out of the room") and a long video are input into a system, and the output is the temporal start and end points which correspond to when the text query occurs in the video. We will detail the collection of the Distinct Describable Moments (DiDeMo) dataset and the

TEMPOral Reasoning in Video and Language (TEMPO) datasets for this task. We then introduce the first models to tackle the task of moment localization in video with natural language.

In recent years the AI community has made tremendous progress on tasks at the intersection of language and vision: image description is, at this point, a mature sub-field. This thesis improves upon prior image description models by introducing models which describe novel objects and produce less biased captions. Beyond image description, we introduce the first deep textual explanation system as well as the new task of moment localization. In the conclusion (Chapter 7), we will discuss how contributions from this thesis could be expanded upon for more complex tasks at the intersection of language and vision.

Chapter 2

Compositional Captioning for Describing Novel Objects

2.1 Problem Statement

Early deep recurrent neural network models achieved promising results on the task of generating descriptions for images and videos [Vin+15; Don+15; KSZ14; KFF15]. Large corpora of paired images and descriptions, such as MSCOCO [Che+15] and Flickr30k [You+14] played an important role in contributing to the success of these methods. However, these datasets describe a relatively small variety of objects in comparison to the number of labeled objects in object recognition datasets, such as ImageNet [Rus+15]. Consequently, though modern object recognition systems have the capacity to recognize thousands of object classes, existing state-of-the-art caption models lack the ability to form *compositional* structures which integrate new objects with known concepts without explicit examples of image-sentence pairs. To address this limitation, we proposed the task of novel object captioning in which models learn to combine visual groundings of lexical units to generate descriptions about objects which are not present in caption corpora (paired image-sentence data), but are present in object recognition datasets (unpaired image data) and text corpora (unpaired text data). By learning to exploit external sources of unpaired data, our description models are able to accurately caption hundreds of new objects. [1]

In contrast to generic description models, our aim is to build *compositional* models in the sense that they can seamlessly construct sentences about new objects by combining them with already seen linguistic expressions in paired training data. To illustrate, consider the image of the otter in Figure 2.1. To describe the image accurately, any captioning model needs to identify the constituent visual elements such as “otter”, “water” and “sitting” and combine them to generate a coherent sentence. While previous deep caption models learn to combine visual elements into a cohesive description exclusively from image and caption pairs, our models can compose a caption to describe a new visual element such as the “otter” by understanding that “otters” are similar to

[1] This chapter is based on joint work done with Subhashini Venugopalan, Marcus Rohrbach, Kate Saenko, Ray Mooney, and Trevor Darrell [Hen+16a; Ven+17] presented at CVPR 2016 and CVPR 2017.

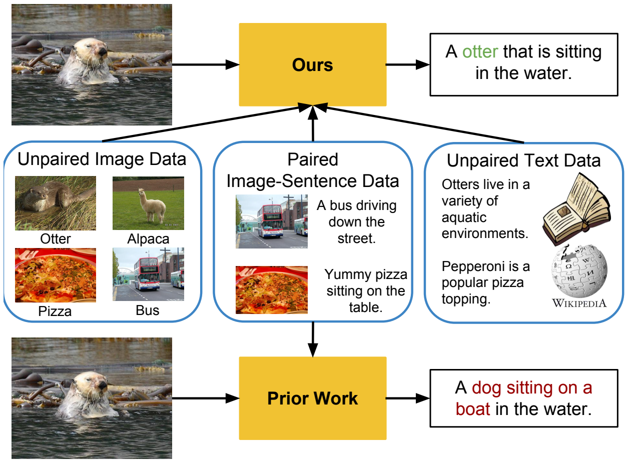

Ours
A otter that is sitting in the water.
Unpaired Image Data
Otter
Alpaca
Pizza
Bus
Paired Image-Sentence Data
A bus driving down the street.
Yummy pizza sitting on the table.
Unpaired Text Data
Otters live in a variety of aquatic environments.
Pepperoni is a popular pizza topping.
Wikipedia
Prior Work
A dog sitting on a boat in the water.

Figure 2.1: Conventional deep caption methods are unable to generate sentences about objects unseen in caption corpora (like otter). In contrast, we propose to effectively incorporates information from independent image datasets and text corpora to compose descriptions about novel objects without any paired image-captions.

“animals” and can thus be composed in the same way with other lexical expressions. To effectively describe new objects, we leverage external text corpora to relate novel objects to concepts seen in paired data and transfer knowledge from object recognition datasets to the description task. In this chapter, we will first describe the Deep Compositional Captioner (DCC) [Hen+16a], the first deep description model capable of describing novel objects. We introduce the novel object captioning split of the MSCOCO description dataset as well as propose metrics to validate our approach. Additionally, we show that our models can be extended to caption images in ImageNet, for which no description data exists. We then discuss a major shortcoming of the DCC model: it is not end-to-end trainable. To mitigate this shortcoming we present the Novel Object Captioner (NOC) [Ven+17] which is end-to-end trainable and achieves superior results on the novel object captioning task when compared to DCC.

2.2 Model: Deep Compositional Captioner

Although it is common to transfer pre-trained weights from image classification tasks trained on unpaired image data to deep caption models, this alone does not allow models to describe objects in unpaired image data. Unlike existing models, we are able to describe objects present in unpaired image data but not present in paired image-sentence data. We do this by transferring knowledge from unpaired image data and additionally, to enhance the language structure, we train our model on independent text corpora. Further, we explore methods to transfer knowledge between semantically related words to compose descriptions of new objects. Our method consists of three stages: 1) training a deep lexical classifier and deep language model with unpaired data, then, 2) com-

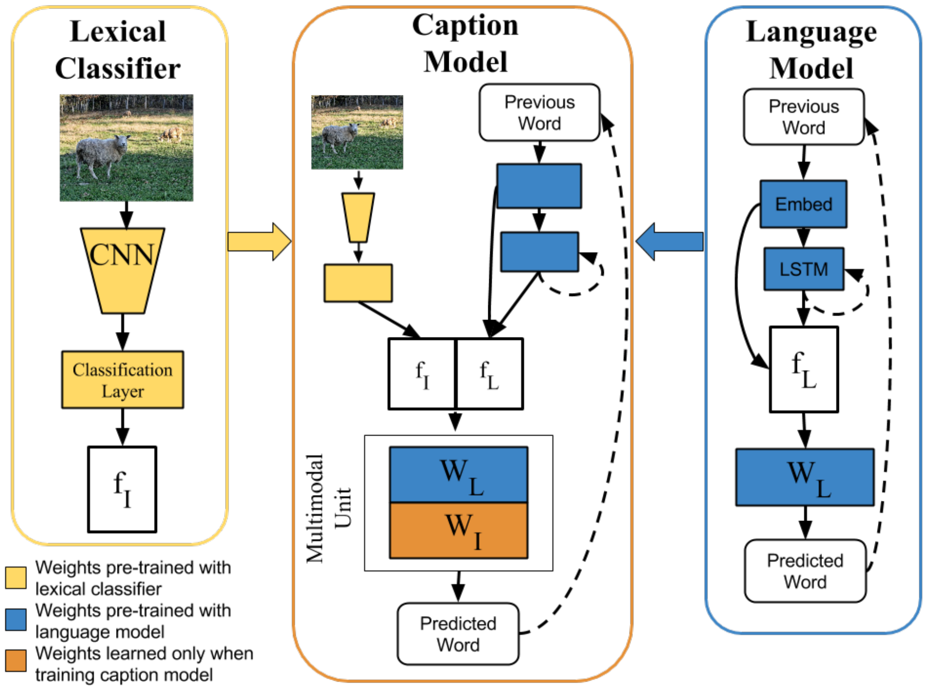


Figure 2.2: DCC consists of a lexical classifier, which maps pixels to semantic concepts and is trained only on unpaired image data, and a language model, which learns the structure of natural language and is trained on unpaired text data. The multimodal unit of DCC integrates the lexical classifier and language model and is trained on paired image-sentence data.

bining the lexical classifier and language model into a caption model which is trained on paired image-sentence data, and, finally, 3) transferring knowledge from words which appear in paired image-sentence data to words which do not appear in paired image-sentence data.

2.2.1 Deep Lexical Classifier

The lexical classifier (Fig 2.2, left) is a CNN which maps images to semantic concepts. In order to train the lexical classifier, we first mine concepts which are common in paired image-text data by extracting the part-of-speech of each word [Tou+03] and then select the most common adjectives, verbs, and nouns. We do not refine the mined concepts, which means some of the concepts, such as “use”, are not strictly visual. In addition to concepts common in paired image-sentence data, the classifier is also trained on objects that we wish to describe outside of the caption datasets.

The lexical classifier is trained by fine-tuning a CNN which is pre-trained on the training split of the ILSVRC-2012 [Rus+15] dataset. When describing images, multiple visual concepts from the image influence the description. For example, the sentence “An alpaca stands in the green grass.” includes the visual concepts “alpaca”, “stands”, “green”, and “grass”. In order to apply multiple labels to each image we use a sigmoid cross-entropy loss over image labels. We denote the image feature output by the lexical classifier as f_I, where each index of f_I corresponds to the probability that a particular concept is present in the image. Our idea of learning visual classifiers from text descriptions for captioning is similar to [RRS15] who learn classifiers for objects, verbs, and locations and [Fan+15] who learn visual concepts using multiple instance learning.

2.2.2 Language Model

The language model (Fig 2.2, right) learns sentence structure using only unpaired text data and includes an embedding layer which maps a one-hot-vector word representation to a lower dimensional space, an LSTM [HS97], and a word prediction layer. The language model is trained to predict a word given previous words in a sentence. At each time step, the previous word is input into the embedding layer. The embedded word is input into an LSTM, which learns the recurrent structure inherent in language. The embedded word and LSTM output are concatenated to form the language features, f_L. f_L is input to an inner product layer which outputs the next word in a generated sequence. At training time, the ground truth word is always used as an input to the language model, but at test time we input the previous word predicted by our model. We also find that results improve by enforcing a constraint that the model cannot predict the same word twice in a row. We explore a variety of sources for unpaired text corpora as described in Section 2.3.1.

2.2.3 Caption Model

The caption model integrates the lexical classifier and the language model to learn a joint model for image description. As shown in Fig 2.2 (center) the multimodal unit in the caption model combines the image features, f_I and the language features, f_L. The multimodal unit we use is an affine transformation of the image and language features:

$$p_w = \text{softmax}(f_I W_I + f_L W_L + b) \tag{2.1}$$

where W_I, W_L, and b are learned weight matrices and p_w is a probability distribution over the predicted word.

Intuitively, the weights in W_I learn to predict a set of words which are likely to occur in a caption given the visual elements discerned by the lexical classifier. In contrast, W_L learns the sequential structure of natural language by learning to predict the next word in a sequence given the previous words. By summing $f_I W_I$ and $f_L W_L$, the multimodal unit combines the visual information learned by the lexical classifier with the knowledge of language structure learned by the language model to form a coherent description of an image.

Both the language model and caption model are trained to predict a sequence of words, whereas the lexical classifier is trained to predict a fixed set of candidate visual elements for a given image. Consequently, the weights W_L, which map language features to a predicted word are learned when training the language model, but the weights W_I are not. Weights in W_L are pretrained using unpaired text data before fine-tuning with paired image-sentence data, W_I are trained purely with image-sentence data. Though we use a linear multimodal unit, our results are comparable to results achieved by other methods which include a nonlinear layer for word prediction. For example, on the MSCOCO validation set [Don+15] achieves a METEOR score of 23.7, and DCC achieves a METEOR of 23.2.

The caption model is designed to enable easy transfer of learned weights from words which appear in the paired image-sentence data to words which do not appear in the image-sentence data. First, by using a lexical classifier to extract image features, image features have explicit semantic meaning. Consequently, it is trivial to expand the image feature to include new objects and to adjust weights in the multimodal unit which correspond to specific objects. Second, by learning language features using unpaired text data, we ensure that the model learns a good embedding for words which are not present in paired image-sentence data. Finally, by using a single-layer, linear multimodal unit, the dependence between image and language features and predicted words is straightforward to understand and easy to exploit for semantic transfer.

2.2.4 Transferring Information Between Objects

Direct Transfer The first method we explore to transfer weights between objects directly transfers learned weights in W_I, W_L and b from words that appear in the paired image-sentence dataset to words which do not appear in a paired image-sentence dataset (Fig 2.3). Intuitively, the direct transfer model requires that a new word is described in the same way that semantically similar words are described. To illustrate, consider the new word "alpaca" which is semantically close to the known word "sheep". Let v_a and v_s indicate the index of the words alpaca and sheep in the vocabulary. Given image and language features, f_I and f_L respectively, the probability of predicting the word "sheep" is proportional to:

$$f_I W_I[:, v_s] + f_L W_L[:, v_s] + b[v_s] \tag{2.2}$$

In order to construct sentences with "alpaca" in the same way sentences are constructed with the word "sheep", we first directly transfer the weights $W_I[:, v_s]$, $W_L[:, v_s]$, and $b[v_s]$ (indicated in red in Fig 2.3) to $W_I[:, v_a]$, $W_L[:, v_a]$, and $b[v_a]$ (indicated in green in Fig 2.3). Additionally, we expect the prediction of the word "sheep" to be highly dependent on the likelihood that a "sheep" is present in the image. In other words, we expect $W_I[:, c_s]$ to strongly weight the output of the lexical classifier which corresponds to the word "sheep". However, $W_I[:, c_a]$ should strongly weight the lexical classifier which corresponds to the word "alpaca". To enforce this, we set $W_I[r_a, c_a] = W_I[r_s, c_s]$ where r_a and r_s indicate the index in the image features which correspond to the alpaca and sheep classifiers respectively. Finally, we do not expect the output of the word "alpaca" to depend on the presence of a sheep in the image and vice versa. Consequently, we set $W_I[r_s, c_a] = W_I[r_a, c_s] = 0$.

Delta Transfer Instead of directly transferring weights, we can also transfer *how weights change* when trained on paired image-text data. Again, consider transferring the word "sheep" to the word "alpaca". We determine Δ_L for a given word as:

$$\Delta_L = W_{L-caption}[:, v_s] - W_{L-language}[:, v_s] \tag{2.3}$$

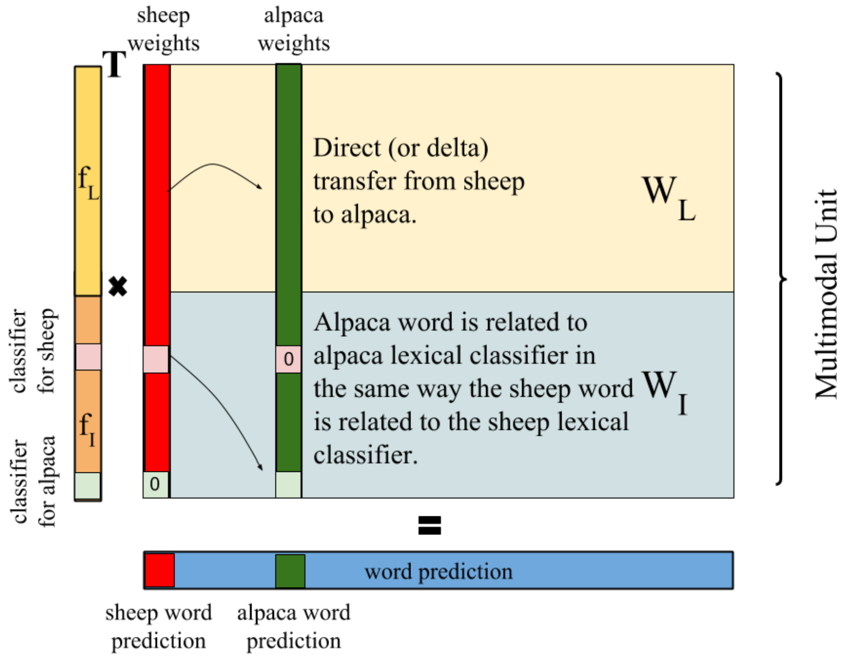

sheep weights
alpaca weights
T
f_L
f_I
×
classifier for sheep
classifier for alpaca
Direct (or delta) transfer from sheep to alpaca.
W_L
Alpaca word is related to alpaca lexical classifier in the same way the sheep word is related to the sheep lexical classifier.
W_I
0
0
Multimodal Unit
=
word prediction
sheep word prediction
alpaca word prediction

Figure 2.3: Method for transferring knowledge from words trained with paired image-sentence data to words trained without image-sentence data. See Section 2.2.4 for details.

where $W_{L-caption}$ are weights learned when training with both images and sentences and $W_{L-language}$ are weights learned when training only with language. The weights for the new word "alpaca" are updated as:

$$W_{L-caption}[:, v_a] = W_{L-language}[:, v_a] + \Delta_L \tag{2.4}$$

Delta transfer may be advantageous because, unlike direct transfer, it does not overwrite pre-trained weights in W_L during transfer. When performing delta transfer for W_L, we still use direct transfer for weights in W_I.

Determining Concept Similarity Determining which words in the paired image-sentence data are semantically similar to words out of the paired image-sentence data is key for transfer. We determine semantic similarity with the word2vec [Mik+13] CBOW model which we trained on the British National Corpus (BNC), UkWaC, and Wikipedia, and estimate word similarity using cosine distance. Additionally, we restrict words that are transferred to new words to be in the lexical layer.

2.3 Experimental Framework

2.3.1 Image Description

Datasets To empirically evaluate our method we create a subset of the MSCOCO [Che+15] training set (denoted as the held-out MSCOCO training set) which excludes all image-sentence pairs which describe at least one of eight MSCOCO objects. To ensure that excluded objects are at least similar to some included ones, we cluster the 80 objects annotated in the MSCOCO segmentation challenge using the vectors from the word2vec embedding described in Section 2.2.4 and exclude one object from each cluster. The following words are chosen: “bottle”, “bus”, “couch”, “microwave”, “pizza”, “racket”, “suitcase”, and “zebra”. We randomly select 50% of the MSCOCO validation set for validation, and set aside the other 50% for test. We label the visual concepts in each image based on the five ground truth caption annotations provided in the MSCOCO dataset. If any of the ground truth captions mentions an object, the corresponding image is considered a positive example for that object.

In addition to empirically evaluating our model, we also qualitatively examine the performance on images containing objects outside of the paired image-sentence corpora. Specifically, we choose over 500 objects from the ImageNet 7k object recognition dataset [Den+10]. We use 75% of images from each class to train the lexical classifier, and evaluate on the rest. We stress that we do not have any descriptions for these categories.

Training the Lexical Classifier We consider both MSCOCO and ImageNet as sources of labeled image data to train the lexical classifier. For all objects in paired image-sentence data, we use COCO images which are labeled with 471 visual concepts to train the lexical classifier. For the eight objects which do not appear in the paired image-sentence data, we explore training the lexical classifier using MSCOCO images (in-domain) and ImageNet images (out-of-domain). For qualitative experiments on ImageNet objects, we use Imagenet images to train the lexical classifier on objects unseen in the paired image-sentence data. The lexical classifier is trained by fine-tuning a deep convolutional model (VGG [Jia+14]) trained on the ILSVRC-2012 [Rus+15] object recognition training subset of the ImageNet dataset.

Training the Language Model We consider three different sources for unpaired text data to train the language model:

1. **MSCOCO** consists of all captions from the MSCOCO train set
2. **External text (WebCorpus)** consists of 60 million sentences from the British National Corpus (BNC), UkWaC, and Wikipedia.
3. **Text from Image Description Corpora (CaptionTxt)** consists of text data from other paired image and video description datasets: Flickr1M [HL08], Flickr30k [PYH14], Pascal-1k [Ras+10] and ImageCLEF-2012 [TP12] and sentence descriptions of Youtube clips from the MSVD training corpus. This corpora *does not* include sentences from MSCOCO.

Training the Caption Model After training the lexical classifier and language model, the weights in the multimodal layer of the caption model are trained with paired image-sentence data. For the delta transfer method, if weights in W_L, which are transferred from the language model, diverge too much from their original values, transfer does not work well. Consequently, we first hold weights in W_L constant, training only W_I, before jointly training W_L and W_I.

2.3.2 Video Description

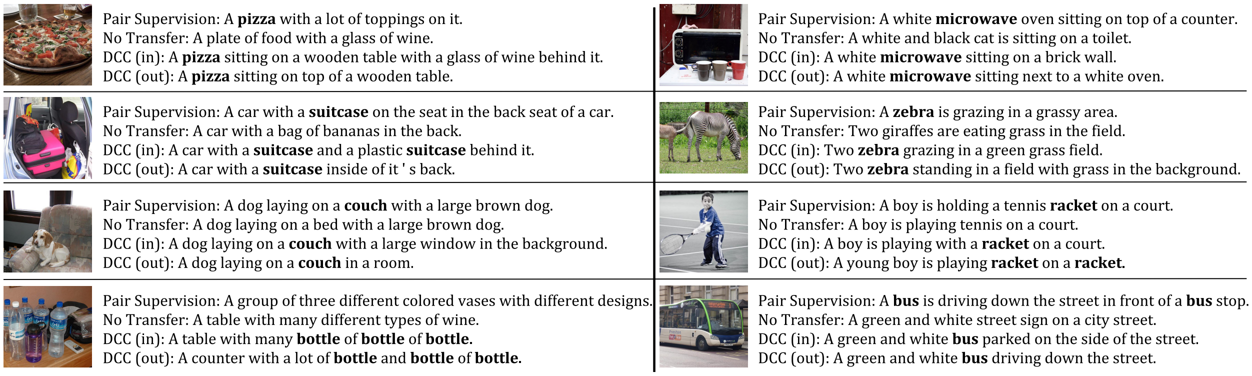
The ability to integrate outside knowledge is even more important in scenarios in which data is harder to collect, such as video description. We therefore apply DCC to video description as well. Our video description experiments closely mirror the image description experiments, though deviate slightly as outlined below.

Datasets For video description, we use a collection of Youtube clips from the Microsoft Video description (MSVD) corpus [CD11], which contains 1,970 short annotated clips. Our basic experimental setting follows previous video description works [Ven+15b; Ven+15a]. However, we hold out paired video-sentence data for some objects during training. Because there is significant variation in the number of video clips containing each object in the MSVD dataset, we create a held-out training set by selecting objects from the ILSVRC video object detection set, and picking those which appear in five or fewer training videos and at least one test video. Our MSVD held-out set excludes paired video-sentence training data which include the objects “zebra”, “hamster”, “broccoli”, and “turtle”.

We also qualitatively evaluate our method on the ILSVRC object detection challenge videos (initial release) which consists of 1952 video snippets of the 30 objects from the ImageNet object detection challenge on videos. Objects which we describe in the ImageNet videos include “whale”, “fox”, “hamster”, “lion”, “zebra”, and “turtle”.

Lexical Classifier Unlike images, videos consist of a sequence of frames which need to be mapped to a set of semantic concepts by the lexical classifier. To build a lexical classifier for videos, we mean-pool fc_7 features across all frames in a video clip before classification. We use both MSVD and ImageNet videos to train the lexical classifier. We use the VGG-16 layer model (from the *Caffe* model zoo) to extract fc_7 layer features from video frames.

Training the Language Model To study the effect of in-domain and out-of-domain language training we use two different text corpora to train the video language model. The first is the **External text (WebCorpus)** described previously. For in-domain text, we consider captions from MSCOCO, Flickr-30k[PYH14], Pascal-1k[Ras+10] and the MSVD sentence descriptions. The caption training and transfer methods are identical for image and video description experiments.



Pair Supervision: A **pizza** with a lot of toppings on it.
No Transfer: A plate of food with a glass of wine.
DCC (in): A **pizza** sitting on a wooden table with a glass of wine behind it.
DCC (out): A **pizza** sitting on top of a wooden table.

Pair Supervision: A white **microwave** oven sitting on top of a counter.
No Transfer: A white and black cat is sitting on a toilet.
DCC (in): A white **microwave** sitting on a brick wall.
DCC (out): A white **microwave** sitting next to a white oven.

Pair Supervision: A car with a **suitcase** on the seat in the back seat of a car.
No Transfer: A car with a bag of bananas in the back.
DCC (in): A car with a **suitcase** and a plastic **suitcase** behind it.
DCC (out): A car with a **suitcase** inside of it ' s back.

Pair Supervision: A **zebra** is grazing in a grassy area.
No Transfer: Two giraffes are eating grass in the field.
DCC (in): Two **zebra** grazing in a green grass field.
DCC (out): Two **zebra** standing in a field with grass in the background.

Pair Supervision: A dog laying on a **couch** with a large brown dog.
No Transfer: A dog laying on a bed with a large brown dog.
DCC (in): A dog laying on a **couch** with a large window in the background.
DCC (out): A dog laying on a **couch** in a room.

Pair Supervision: A boy is holding a tennis **racket** on a court.
No Transfer: A boy is playing tennis on a court.
DCC (in): A boy is playing with a **racket** on a court.
DCC (out): A young boy is playing **racket** on a **racket.**

Pair Supervision: A group of three different colored vases with different designs.
No Transfer: A table with many different types of wine.
DCC (in): A table with many **bottle** of **bottle** of **bottle.**
DCC (out): A counter with a lot of **bottle** and **bottle** of **bottle.**

Pair Supervision: A **bus** is driving down the street in front of a **bus** stop.
No Transfer: A green and white street sign on a city street.
DCC (in): A green and white **bus** parked on the side of the street.
DCC (out): A green and white **bus** driving down the street.

Figure 2.4: Image Description: Comparison of captions generated by model without transfer, DCC with in-domain training (MSCOCO), with out-of-domain training (ImageNet and WebCorpus), and a model trained with paired image-sentence supervision for all MSCOCO objects. DCC is capable of integrating new words and generates sentences similar to those generated when paired image-sentences for all objects are present during training.

2.3.3 Metrics

To evaluate our transfer methods, we must choose a metric that indicates whether or not a generated sentence includes a new object. Common caption metrics such as BLEU [Pap+02] and METEOR [BL05] measure overall sentence meaning and fluency. However, for many objects, it is possible to achieve good BLEU and METEOR scores without mentioning the new object (e.g., consider sentences describing the boy playing tennis in Figure 2.4). To definitively report our models ability to integrate new vocabulary, we also report the F1-score. The F1-score considers "false positives" (when a word appears in a sentence it should not appear in), "false negatives" (when a word does not appear in a sentence it should appear in), and "true positives" (when a word appears in a sentence it should appear in). We consider generated sentences "positive" if they contain at least one mention of a held out word and ground truth sentences "positive" if a word is mentioned in any ground truth annotation that describes an image.

Our models are trained using *Caffe*[Jia+14] and are publicly released [2].

2.4 Results: Deep Compositional Captioner

2.4.1 Image Description

As shown in Figure 2.4, DCC is capable of integrating new vocabulary into image descriptions in a cohesive manner. We first empirically analyze and ablate our model on the MSCOCO held out set, then consider describing ImageNet images.

[2] https://people.eecs.berkeley.edu/~lisa_anne/dcc_project_page.html

	LRCN [Don+15]	No Transfer	ΔT	DT
F1	0	0	34.89	**39.78**
BLEU-1	63.65	62.99	64.00	**64.40**
METEOR	19.33	19.9	20.86	**21.00**

Table 2.1: We compare DCC before transfer (No Transfer) to DCC with delta transfer (ΔT) and DCC with direct transfer (DT). We also compare to another competitive caption generation model (LRCN). We measure our models ability to insert new words into a generated sentence with the F1-score. We also report Bleu-1 and METEOR, which indicates overall sentence quality. DCC successfully incorporates new words and improves sentence quality. (Values in %)

Direct Transfer Versus Delta Transfer Table 2.1 compares the average F1-score across the eight held-out training classes for the delta transfer and direct transfer methods. We additionally train LRCN ([Don+15]) [3] on our MSCOCO held-out dataset and note that our model without transfer yields comparable results to LRCN, and performs considerably better on all metrics after transfer. As shown by the F1-scores reported in Table 2.1, both the delta transfer and direct transfer methods are capable of integrating new words into their vocabulary. We also report the BLEU-1 score, which measures the overlap between generated words and words in reference sentences. By measuring the METEOR score, we ensure that our model maintains sentence fluency when inserting new objects. DCC consistently increases METEOR scores indicating that overall sentence quality improves with DCC. The direct transfer method improves F1-scores, BLEU, and METEOR scores by a larger amount than the delta transfer method and is thus used for the remainder of our experiments.

Importantly, BLEU and METEOR scores do not decrease for objects which are present in the held-out training data set. When trained with all image-sentence training examples, our model achieves an average BLEU-1 of 69.36 and METEOR of 23.98 on held-out classes.

To illustrate which words our model works best on, we report the F1-score for individual objects in Table 2.2. We compare to a model which is trained with image-sentence pairs for the eight held-out objects. For all objects, DCC is able to compose sentences which include the object. We notice that DCC tends to do better for objects which might be more prominent in a scene (such as “zebra” or “couch”), and does worse on smaller objects like “bottle”.

Analysis of Transfer Words In general, determining word similarity with a word2vec embedding works well. Words such as “zebra”/“giraffe” and “microwave”/“refrigerator” are close in embedding space and are also used in similar ways in natural language, suggesting they will work well for transfer. Some transfer pairs (“racket”/“tennis” and “bus”/“stop”) are used together frequently but play different structural roles in sentences. Consequently, the word “racket” is frequently used like the word “tennis” leading to grammatical errors. However, similar errors do not occur when transferring “stop” to “bus”.

[3] For fair comparison, we train LRCN on VGG, fine-tune through the entire network, and do not use beam search.

	bottle	bus	couch	microwave	pizza	racket	suitcase	zebra	average
Pair Sup.	23.20	72.07	50.60	39.48	77.07	38.52	46.50	91.02	54.81
DT	4.63	29.79	45.87	28.09	64.59	52.24	13.16	79.88	39.78

Table 2.2: Image Description: Comparison of F1 scores for direct transfer DCC model (DT) and a pair supervision (Pair Sup.) model trained with image-sentence training examples for all objects. (Values in %)

Lexical classifier	Language model	B-1	METEOR	F1
MSCOCO	MSCOCO	64.40	21.00	39.78
Imagenet	MSCOCO	64.00	20.71	33.60
Imagenet	CaptionTxt	64.79	20.66	35.53
Imagenet	WebCorpus	64.85	20.66	34.94

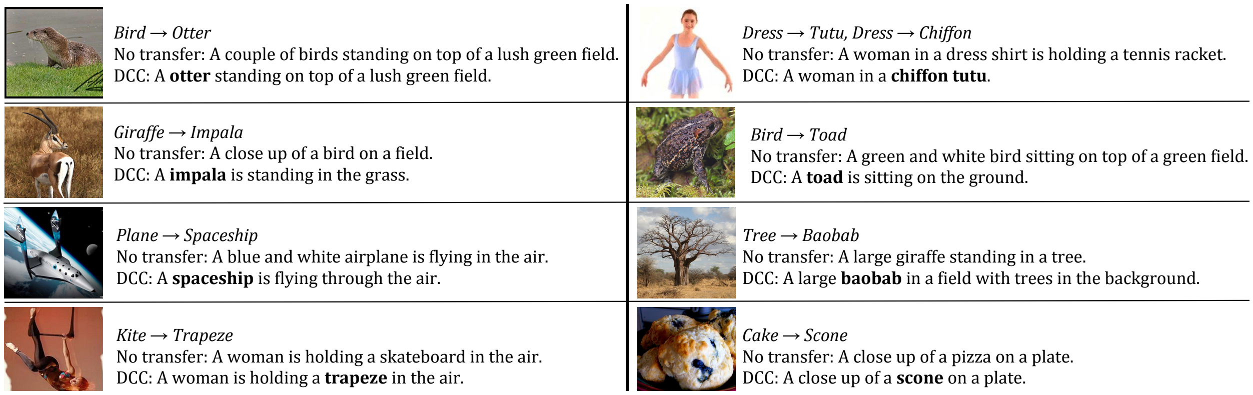
Table 2.3: Image Description: We compare the effect of pre-training the lexical classifier and language model with different unpaired image and text data sets. As expected, we see the best result when using in domain MSCOCO data to train the lexical classifier and language model, though training with out of domain corpora is comparable. (Values in %)

Pre-Training with Out-of-Domain Data In the above experiments the lexical classifier and language model are pre-trained using MSCOCO images and text. In a real world scenario, it is unlikely that available unpaired image and text data will be from the same domain as paired image-sentence data. However, it is essential that the model learns good image and language features. Naturally, if the lexical classifier is unable to recognize certain objects, DCC will not be able to describe the objects. Perhaps more subtly, if the language model is not trained with unpaired text which includes an object, it will not learn a proper embedding for the new word and will not produce cohesive descriptions about new objects.

Table 2.3 demonstrates the impact of using outside image and text corpora to train the lexical classifier and language model. Our model performs best when provided with in-domain image and text for all training stages, but performance is comparable when using ImageNet images to train the lexical classifier and CaptionTxt or WebCorpus text data to train the language model.

2.4.2 Describing ImageNet Objects

We qualitatively assess our model by describing a variety of ImageNet objects which are not included in the MSCOCO data set (Fig 2.5). DCC accurately describes 335 new words including entry-level words like “toad” as well as fine-grained categories like “baobab”. Though most Imagenet words we transfer are nouns, we are able to successfully transfer some adjectives such as “chiffon”. DCC achieves more than simple noun replacement. For example, the sentence “A large



Bird → Otter
No transfer: A couple of birds standing on top of a lush green field.
DCC: A **otter** standing on top of a lush green field.

Dress → Tutu, Dress → Chiffon
No transfer: A woman in a dress shirt is holding a tennis racket.
DCC: A woman in a **chiffon tutu**.

Giraffe → Impala
No transfer: A close up of a bird on a field.
DCC: A **impala** is standing in the grass.

Bird → Toad
No transfer: A green and white bird sitting on top of a green field.
DCC: A **toad** is sitting on the ground.

Plane → Spaceship
No transfer: A blue and white airplane is flying in the air.
DCC: A **spaceship** is flying through the air.

Tree → Baobab
No transfer: A large giraffe standing in a tree.
DCC: A large **baobab** in a field with trees in the background.

Kite → Trapeze
No transfer: A woman is holding a skateboard in the air.
DCC: A woman is holding a **trapeze** in the air.

Cake → Scone
No transfer: A close up of a pizza on a plate.
DCC: A close up of a **scone** on a plate.

Figure 2.5: Image Description: DCC is able to describe Imagenet objects (bolded) which are not mentioned in any of the paired image-sentence data, and therefore cannot be described by existing deep caption models. X → Y indicates that the known word X is transferred to the new word Y.

giraffe standing in a tree" changes significantly to "A large baobab in a field with trees in the background" after transfer. Importantly, our model is able to compose sentences by placing objects in the correct context. For example, comparing Fig 2.5 (top left) to the image in Fig 2.1, the object "otter" is correctly described as either "sitting in the water" or "standing on top of a lush green field" depending on visual context.

Figure 2.6 examines a few common error types:

1. **New Object Not Mentioned** (Figure 2.6, top left) For some images, DCC produces relevant sentences, but fails to mention the new object.

2. **Grammatically Incorrect** (Figure 2.6, bottom left) Some sentences incorporate new words, but are grammatically incorrect. For example, though DCC describes sentences with the word "gymnastics", the resulting sentences are frequently grammatically incorrect (e.g., "A woman playing gymnastics on a gymnastics court"). This is likely because the word "tennis" is transferred to "gymnastics". Though both of these words are sports, one does not "play" gymnastics and gymnastics is not performed on a "court."

3. **Object Hallucination** (Figure 2.6, top right) DCC frequently hallucinates objects which commonly occur in a specific visual context. For example, in a beach image, the model commonly includes the word "surfboard".

4. **Irrelevant Description** (Figure 2.6, bottom right) Some captions do not mention any salient objects correctly. Such errors can be caused by poor image recognition or because the language model is unable to construct a reasonable sentence from constituent visual elements.

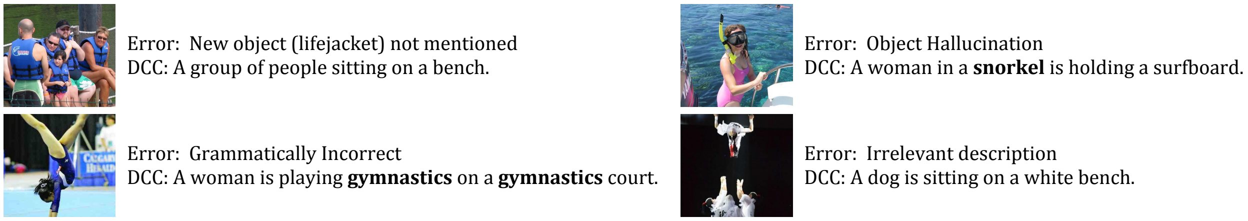

Error: New object (lifejacket) not mentioned
DCC: A group of people sitting on a bench.
Error: Object Hallucination
DCC: A woman in a **snorkel** is holding a surfboard.
Error: Grammatically Incorrect
DCC: A woman is playing **gymnastics** on a **gymnastics** court.
Error: Irrelevant description
DCC: A dog is sitting on a white bench.

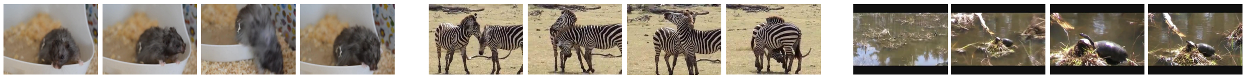
Figure 2.6: Image Description: We highlight four common error types generated by the DCC. See Section 2.4.2 for details.

Model (Video)	METEOR	F1
Baseline (No Transfer)	28.8	0.0
+ DT	28.9	6.0
+ ILSVRC Videos (No Transfer)	29.0	0.0
+ ILSVRC Videos + DT	29.1	22.2

Table 2.4: Video Description: METEOR scores across the test dataset and average F1 scores for the four held-out categories (All values in %) using direct transfer (DT). The DCC models were trained on videos with 4 objects removed and the language model was trained on in-domain sentences.

2.4.3 Video description

We believe DCC can be especially beneficial in domains, such as video description, where the amount of paired training data is small. Table 2.4 presents empirical results of direct transfer DCC on videos in the MSVD corpus (Section 2.3.1). We report the average F1 score on all held-out classes, and METEOR scores on the complete test dataset. As seen by the F1 score, transferring weights allows us to describe new objects in video. Additionally, the METEOR score improves with transfer demonstrating that DCC improves overall sentence quality. Similar to the trend seen for image captioning, training on in-domain text corpora achieves slightly better performance than training on external text. When adding ImageNet videos, both F1 and METEOR increase suggesting that including outside image data is beneficial. Including ImageNet videos to learn better lexical classifiers especially improves the F1 score, which increases from 6.0 to 22.2. Figure 2.7 presents qualitative results of our best model on snippets with the held out objects in MSVD corpus



A **hamster** is eating food in a bowl. A **zebra** is eating some grass. A **turtles** are running.

Figure 2.7: Video Description: Captions generated by DCC on videos of novel objects unseen in paired training data.

and the ILSVRC validation set.

2.4.4 Shortcomings of DCC

With DCC, we are able to describe novel objects in both images and videos – a capability prior work did not have. In order to describe novel objects, we rely on a post-processing step in which we transfer information between semantically related classes. Additionally, in order to ensure that the information learned from unpaired text and unpaired language data is not "forgotten" when training the caption model, we must freeze the weights in the language model and image model. Consequently, lower level language and image model weights are not fine-tuned to the captioning task. This causes two problems. First, our model is not end-to-end trainable. General wisdom in the deep learning community is that when enough data is available, end-to-end trainable models perform substantially better [Yos+14]. For example, in LRCN [Don+15] we observed that freezing CNN weights and only training the LSTM with paired image-sentence data resulted in poorer performance than fine-tuning the entire network with image-sentence data. Second, though our model can describe novel objects, it is unclear how it can be adapted to describe *rare* objects. Like novel object captioning, it seems reasonable to believe external text and image data would help when describing objects that appear infrequently in paired image-sentence data. However, if we transfer information from common objects to rare objects in a post-processing step, any information that might be learned about a rare object from image-sentence data would be overwritten. In the next section, we describe a new end-to-end trainable model, the *Novel Object Captioner* [Ven+17], which remedies the shortcomings of DCC.

2.5 Model: Novel Object Captioner

Our NOC model is illustrated in Fig. 2.8. Like DCC, it consists of a language model (Fig. 2.8, right), lexical classifier (Fig. 2.8, left), and caption model (Fig. 2.8, middle). We first pretrain the language model and lexical classifier, and then transfer weights to the caption model, similar to as is done in DCC. To train our description model, instead of freezing the language model and lexical classifier weights as is done in DCC, we introduce auxiliary loss functions (objectives) and tweak the caption model architecture so we can jointly train different components on multiple data sources, such that the model simultaneously learns an independent object recognition model, language model, and caption model. Additionally, our language model leverages distributional semantic embeddings trained on unannotated text *during training* as opposed to in a post-processing step, as is done in DCC. We first discuss the auxiliary objectives and the joint training, and then discuss how we leverage embeddings trained with external text to compose descriptions about novel objects.

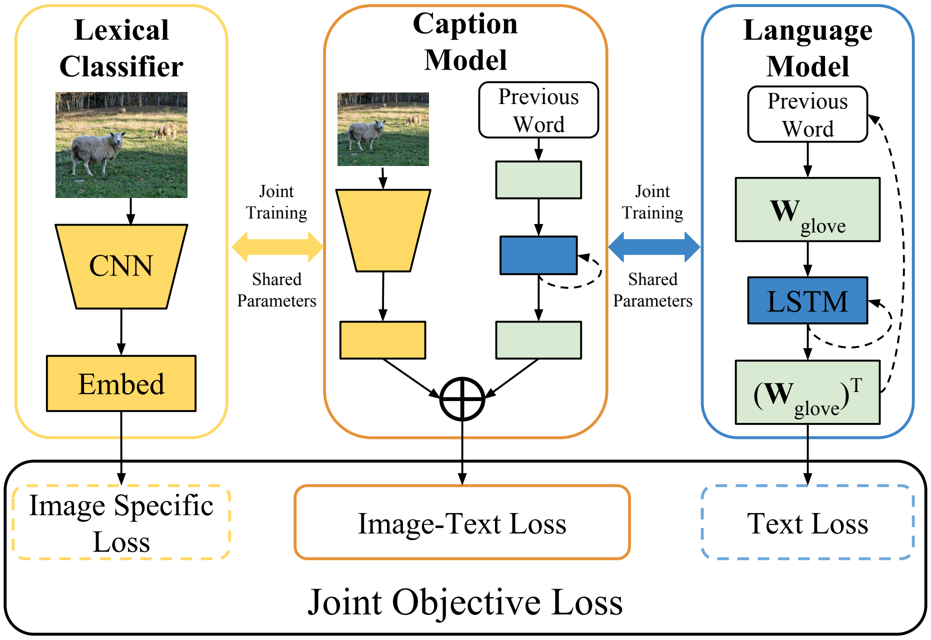


Figure 2.8: Our NOC image caption network. During training, the visual recognition network (left), the LSTM-based language model (right), and the caption model (center) are trained simultaneously on different sources with different objectives but with shared parameters, thus enabling novel object captioning.

2.5.1 Auxiliary Training Objectives

Our motivation for introducing auxiliary objectives is to learn how to describe images without losing the ability to recognize more objects. Typically, image-captioning models incorporate a visual classifier pre-trained on a source domain (e.g. ImageNet dataset) and then tune it to the target domain (the image-caption dataset). However, important information from the source dataset can be suppressed if similar information is not present when fine-tuning, leading the network to forget (over-write weights) for objects not present in the target domain. This is problematic in our scenario in which the model relies on the source datasets to learn a large variety of visual concepts not present in the target dataset. However, with pre-training as well as the complementary auxiliary objectives the model maintains its ability to recognize a wider variety of objects and is encouraged to describe objects which are not present in the target dataset at test time.

2.5.1.1 Image-specific Loss

Our lexical classifier (Fig. 2.8, left) is a neural network parametrized by θ_I and is trained on object recognition datasets. Unlike typical visual recognition models that are trained with a single label on a classification task, for the task of image captioning an image model that has high confidence over multiple visual concepts occurring in an image simultaneously would be preferable. Hence, we choose to train our model using multiple labels with a multi-label loss. As is done in DCC, the lexical classifier is trained to predict multiple visual concepts for each image. Thus to train the lexical classifier, we use a multi-label loss. Unlike DCC, the lexical classifier predicts a score for all words in the vocabulary (including non-visual words like “a” and “the”). If l denotes a label and z_l denotes the binary ground-truth value for the label, then the objective for the visual model is given by the cross-entropy loss ($\mathcal{L}_{IM}$):

$$\mathcal{L}_{\mathcal{IM}}(I;\theta_I) = -\sum_{l}\Big[z_l\, log(S_l(f_{IM}(I;\theta_I))) + (1-z_l)\, log(1-S_l(f_{IM}(I;\theta_I)))\Big] \tag{2.5}$$

where $S_i(x)$ is the output of a $\mathrm{softmax}$ function over index i and input x, and f_{IM}, is the activation of the final layer of the visual recognition network.

2.5.1.2 Text-specific Loss

Our language model (Fig. 2.8, right) is based on LSTM Recurrent Neural Networks, and is trained in the same way as the language model in DCC. We denote the parameters of this network by θ_L, and the activation of the final layer of this network by f_{LM}. The language model is trained to predict the next word w_t in a given sequence of words $w_0, ..., w_{t-1}$. This is optimized using the $\mathrm{softmax}$ loss $\mathcal{L}_{\mathcal{LM}}$ which is equivalent to the maximum-likelihood:

$$\mathcal{L}_{\mathcal{LM}}(w_0, ..., w_{t-1};\theta_L) = -\sum_{t} log(S_{w_t}(f_{LM}(w_0, ..., w_{t-1};\theta_L))) \tag{2.6}$$

2.5.1.3 Image-caption Loss

The goal of the image captioning model (Fig. 2.8, center) is to generate a sentence conditioned on an image (I). NOC predicts the next word in a sequence, w_t, conditioned on previously generated words ($w_0, ..., w_{t-1}$) and an image (I), by summing activations from the deep language model, which operates over previous words, and the deep image model, which operates over an image. We denote these final (summed) activations by f_{CM}. Then, the probability of predicting the next word is given by, $P(w_t|w_0, ..., w_{t-1}, I)$

$$\begin{aligned} &= S_{w_t}(f_{CM}(w_0, ..., w_{t-1}, I;\theta)) \\ &= S_{w_t}(f_{LM}(w_0, ..., w_{t-1};\theta_L) + f_{IM}(I;\theta_I)) \end{aligned} \tag{2.7}$$

Given pairs of images and descriptions, the caption model optimizes the parameters of the underlying language model (θ_L) and image model (θ_I) by minimizing the caption model loss $\mathcal{L}_{\mathcal{CM}}$: $\mathcal{L}_{\mathcal{CM}}(w_0, ., w_{t-1}, I;\theta_L, \theta_I)$

$$= -\sum_{t} log(S_{w_t}(f_{CM}(w_0, ., w_{t-1}, I;\theta_L, \theta_I))) \tag{2.8}$$

While many previous approaches have been successful on image captioning by pre-training the image and language models and tuning the caption model alone (Eqn. 2.8), this is insufficient to generate descriptions for objects outside of the image-caption dataset since the model tends to "forget" (over-write weights) for objects only seen in external data sources. To remedy this, we propose to train the image model, language model, and caption model simultaneously on different data sources. The NOC model's final objective simultaneously minimizes the three individual complementary objectives:

$$\mathcal{L} = \mathcal{L}_{\mathcal{CM}} + \mathcal{L}_{\mathcal{IM}} + \mathcal{L}_{\mathcal{LM}} \tag{2.9}$$

By sharing the weights of the caption model's network with the image network and the language network (as depicted in Fig. 2.8 (a)), the model can be trained simultaneously on independent image-only data, unannotated text data, as well as paired image-caption data. Consequently, co-optimizing different objectives aids the model in recognizing categories outside of the paired image-sentence data.

2.5.2 Language Model with Semantic Embeddings

In DCC, the semantic relationship between words encoded in distributional embeddings [Mik+13; PSM14] was used to transfer weights in the model after training. In NOC, we integrate the semantic relationship between words directly into the language model by transferring a rich word embedding space (Glove [PSM14]) directly into our model. Our language model consists of the following components: a continuous lower dimensional embedding space for words (W_{glove}), a single recurrent (LSTM) hidden layer, and two linear transformation layers where the second layer (W_{glove}^T) maps the vectors to the size of the vocabulary. Specifically, the initial input embedding space (W_{glove}) is used to represent the input (one-hot) words into semantically meaningful dense fixed-length vectors. While the final transformation layer (W_{glove}^T) Finally a softmax activation function is used on the output layer to produce a normalized probability distribution. The cross-entropy loss which is equivalent to the maximum-likelihood is used as the training objective.

2.6 Results: Novel Object Captioner

2.6.1 Experimental Setup: Novel Object Captioner

Our experimental setup to evaluate the NOC closely mirrors the experimental setup for the DCC model. As our external text corpus, we use the WebCorpus from [Hen+16a]. Like DCC, we evaluate the NOC model on image description using the held-out MSCOCO dataset and by generating sentences for ImageNet. Further, to study how well our model can describe rare objects, we pick a separate set of 52 objects which are in ImageNet but mentioned infrequently in MSCOCO (52 mentions on average, with median 27 mentions across all 400k training sentences). We note that it is unclear how to adapt the DCC architecture to the task of describing rare words.

2.6.2 Experiments on MSCOCO

We perform the following experiments to compare NOC's performance with DCC:

Model	bottle	bus	couch	microwave	pizza	racket	suitcase	zebra	Avg. F1	METEOR
DCC	4.63	29.79	**45.87**	**28.09**	64.59	52.24	13.16	79.88	39.78	21.00
NOC	**17.78**	**68.79**	25.55	24.72	**69.33**	**55.31**	**39.86**	**89.02**	**48.79**	**21.32**

Table 2.5: MSCOCO Captioning: F1 scores (in %) of NOC (our model) and DCC [Hen+16a] on held-out objects not seen jointly during image-caption training, along with the average F1 and METEOR scores of the generated captions across images containing these objects.

1. We evaluate the model's ability to caption objects that are held out from MSCOCO during training (Sec. 2.6.2.1).
2. To study the effect of the data source on training, we report performance of when the image and language networks are trained on in-domain and out-of-domain sources (Sec. 2.6.2.2).
3. We perform ablations to study how much each component of our model (such as word embeddings, auxiliary objective, etc.) contributes to the performance (Sec. 2.6.2.3).
4. We also study if the model's performance remains consistent when holding out a different subset of objects from MSCOCO (Sec. 2.6.2.4).

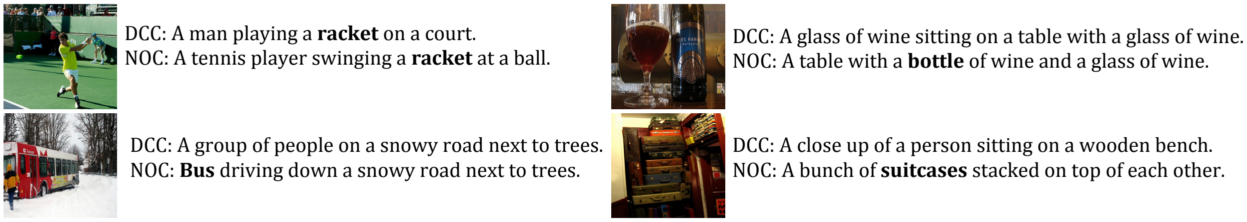
2.6.2.1 Empirical Evaluation on MSCOCO

We empirically evaluate the ability of our proposed model to describe novel objects by following the experimental setup of DCC. We optimize each loss in our model with the following datasets: the caption model, which jointly learns the parameters θ_L and θ_I, is trained only on the subset of MSCOCO without the 8 objects, the image model, which updates parameters θ_I, is optimized using labeled images, and the language model which updates parameters θ_L, is trained using the corresponding descriptions. When training the visual network on images from COCO, we obtain multiple labels for each image by considering all words in the associated captions as labels after removing stop words. We first present evaluations for the in-domain setting in which the image classifier is trained with all COCO training images and the language model is trained with all sentences.

COCO heldout objects. Table 2.5 compares the F1 score achieved by the previous best method, DCC, on the 8 held-out COCO objects. NOC outperforms DCC (by 10% F1 on average) and on all objects except "couch" and "microwave". The higher F1 and METEOR demonstrate that NOC is able to correctly recognize many more instances of the unseen objects and also integrate the words into fluent descriptions.

2.6.2.2 Training data source

To study the effect of different data sources, we also evaluate our model in an out-of-domain setting where classifiers for held out objects are trained with images from ImageNet and the language



DCC: A man playing a **racket** on a court.
NOC: A tennis player swinging a **racket** at a ball.

DCC: A glass of wine sitting on a table with a glass of wine.
NOC: A table with a **bottle** of wine and a glass of wine.

DCC: A group of people on a snowy road next to trees.
NOC: **Bus** driving down a snowy road next to trees.

DCC: A close up of a person sitting on a wooden bench.
NOC: A bunch of **suitcases** stacked on top of each other.

Figure 2.9: COCO Captioning: Examples comparing captions by NOC (ours) and DCC on held out objects from MSCOCO.

	Image	Text	Model	METEOR	F1
1	Baseline (no transfer)		LRCN	19.33	0
			DCC	19.90	0
2	Image Net	Web Corpus	DCC	20.66	34.94
			NOC	17.56	36.50
3	COCO	Web Corpus	NOC	19.18	41.74
4	COCO	COCO	DCC	21.00	39.78
			NOC	**21.32**	**48.79**

Table 2.6: Comparison with different training data sources on 8 held-out COCO objects. Having in-domain data helps both the DCC and our NOC model caption novel objects.

model is trained on text mined from external corpora. Table 2.6 reports average scores across the eight held-out objects. We compare our NOC model to results from DCC, as well as a competitive image captioning model - LRCN [Don+15] trained on the same split. In the out-of-domain setting (line 2), for the chosen set of 8 objects, NOC performs slightly better on F1 and a bit lower on METEOR compared to DCC. However, as previously mentioned, DCC needs to explicitly identify a set of "seen" object classes to transfer weights to the novel classes whereas NOC can be used for inference directly. DCC's transfer mechanism also leads to peculiar descriptions. E.g., *Racket* in Fig. 2.9.

With COCO image training (line 3), F1 scores of NOC improves considerably even with the Web Corpus LM training. Finally in the in-domain setting (line 4) NOC outperforms DCC on F1 by around 10 points while also improving METEOR slightly. This suggests that NOC is able to associate the objects with captions better with in-domain training, and the auxiliary objectives and embedding help the model to generalize and describe novel objects.

Contributing factor	Glove	LM pretrain	Tuned Visual Classifier	Auxiliary Objective	METEOR	F1
Tuned Vision	-	-	✓	✓	15.78	14.41
LM & Embedding	✓	✓	✓	-	19.80	25.38
LM & Pre-trained Vision	✓	✓	Fixed	-	18.91	39.70
Auxiliary Objective	✓	-	✓	✓	19.69	47.02
All	✓	✓	✓	✓	**21.32**	**48.79**

Table 2.7: Ablations comparing the contributions of the Glove embedding, LM pre-training, and auxiliary objectives, of the NOC model. Our auxiliary objective along with Glove have the largest impact in captioning novel objects.

Model	bed	book	carrot	elephant	spoon	toilet	truck	umbrella	Avg. F1	METEOR
NOC	53.31	18.58	20.69	85.35	2.70	73.61	57.90	54.23	45.80	20.04

Table 2.8: MSCOCO Captioning: F1 scores (in %) of NOC (our model) on a different subset of the held-out objects not seen jointly during image-caption training, along with the average F1 and METEOR scores of the generated captions across images containing these objects. NOC is consistently able to caption different subsets of unseen object categories in MSCOCO.

2.6.2.3 Ablations

Table 2.7 compares how different aspects of training impact the overall performance.

Tuned Vision contribution The model that does not incorporate Glove or LM pre-training has poor performance (METEOR 15.78, F1 14.41); this ablation shows the contribution of the vision model alone in recognizing and describing the held out objects.

LM & Glove contribution: The model trained without the auxiliary objective, performs better with F1 of 25.38 and METEOR of 19.80; this improvement comes largely from the GloVe embeddings which help in captioning novel object classes.

LM & Pre-trained Vision: It's interesting to note that when we fix classifier's weights (pre-trained on all objects), before tuning the LM on the image-caption COCO subset, the F1 increases substantially to 39.70 suggesting that the visual model recognizes many objects but can "forget" objects learned by the classifier when fine-tuned on the image-caption data (without the 8 objects).

Auxiliary Objective: Incorporating the auxiliary objectives, F1 improves remarkably to 47.02. We note here that by virtue of including auxiliary objectives the visual network is tuned on all images thus retaining it's ability to classify/recognize a wide range of objects. Finally, incorporating all aspects gives NOC the best performance (F1 48.79, METEOR 21.32), significantly outperforming DCC.

2.6.2.4 Validating on a different subset of COCO

To show that our model is consistent across objects, we create a different training/test split by holding out a different set of eight objects from COCO. The objects we hold out are: bed, book, carrot, elephant, spoon, toilet, truck and umbrella. Images and sentences from these eight objects again constitute about 10% of the MSCOCO training dataset. Table 2.8 presents the performance of the model on this subset. We observe that the F1 and METEOR scores, although a bit lower, are consistent with numbers observed in Table 2.5 confirming that our model is able to generalize to different subsets of objects.

2.6.3 Scaling to ImageNet

To demonstrate the scalability of NOC, we describe objects in ImageNet for which no paired image-sentence data exists. Our experiments are performed on two subsets of ImageNet, (i) Novel Objects: A set of 638 objects which are present in ImageNet as well as the model's vocabulary but are not mentioned in MSCOCO. (ii) Rare Objects: A set of 52 objects which are in ImageNet as well as the MSCOCO vocabulary but are mentioned infrequently in the MSCOCO captions (median of 27 mentions). For quantitative evaluation, (i) we measure the percentage of objects for which the model is able to describe at least one image of the object (using the object label), (ii) we also report accuracy and F1 scores to compare across the entire set of images and objects the model is able to describe. Furthermore, we obtain human evaluations comparing DCC to NOC on whether the model is able to incorporate the object label meaningfully in the description together with how well it describes the image.

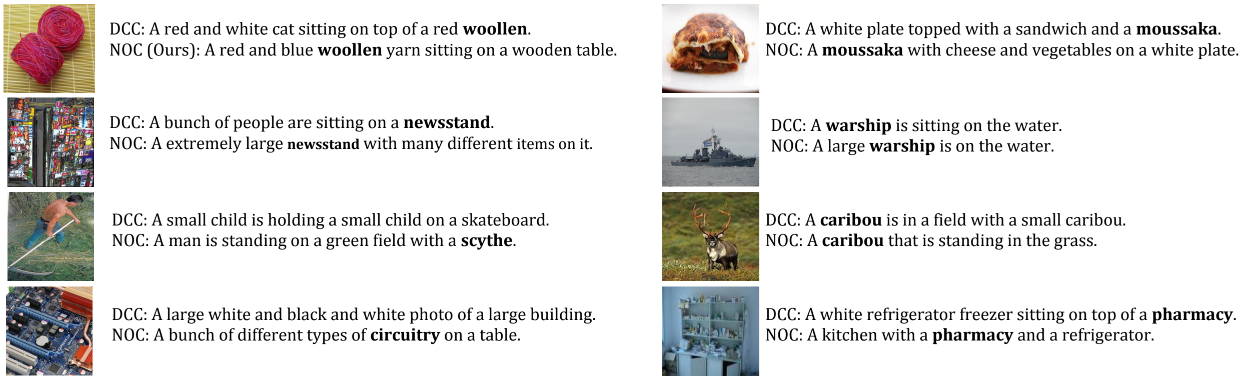
2.6.3.1 Describing Novel Objects

Table 2.9 compares models on 638 novel object categories (identical to DCC) using the following metrics: (i) Describing novel objects (%) refers to the percentage of the selected ImageNet objects mentioned in descriptions, i.e. for each novel word (e.g., "otter") the model should incorporate the word ("otter") into at least one description about an ImageNet image of the object (otter). While DCC is able to recognize and describe 56.85% (363) of the selected ImageNet objects in descriptions, NOC recognizes several more objects and is capable of describing 91.27% (582 of 638) ImageNet objects. (ii) Accuracy refers to the percentage of images from each category where the model is able to correctly identify and describe the category. We report the average accuracy across all categories. DCC incorporates a new word correctly 11.08% of the time, in comparison, NOC improves this appreciably to 24.74%. (iii) F1 score is computed based on precision and recall of mentioning the object in the description. Again, NOC outperforms with average F1 33.76% to DCC's 14.47%.

Although NOC and DCC use the same CNN, NOC is both able to describe more categories, and correctly integrate new words into descriptions more frequently. DCC can fail either with respect to finding a suitable object that is both semantically and syntactically similar to the novel object, or with regard to their language model composing a sentence using the object name, in

Model	Desc. Novel (%)	Acc (%)	F1 (%)
DCC	56.85	11.08	14.47
NOC	**91.27**	**24.74**	**33.76**

Table 2.9: ImageNet: Comparing our model against DCC on % of novel classes described, average accuracy of mentioning the class in the description, and mean F1 scores for object mentions.



DCC: A red and white cat sitting on top of a red **woollen**.
NOC (Ours): A red and blue **woollen** yarn sitting on a wooden table.

DCC: A white plate topped with a sandwich and a **moussaka**.
NOC: A **moussaka** with cheese and vegetables on a white plate.

DCC: A bunch of people are sitting on a **newsstand**.
NOC: A extremely large **newsstand** with many different items on it.

DCC: A **warship** is sitting on the water.
NOC: A large **warship** is on the water.

DCC: A small child is holding a small child on a skateboard.
NOC: A man is standing on a green field with a **scythe**.

DCC: A **caribou** is in a field with a small caribou.
NOC: A **caribou** that is standing in the grass.

DCC: A large white and black and white photo of a large building.
NOC: A bunch of different types of **circuitry** on a table.

DCC: A white refrigerator freezer sitting on top of a **pharmacy**.
NOC: A kitchen with a **pharmacy** and a refrigerator.

Figure 2.10: ImageNet Captioning: Examples comparing captions by NOC and DCC on objects from ImageNet.

NOC the former never occurs (i.e. we don't need to explicitly identify similar objects), reducing the overall sources of error.

Fig. 2.10 and Fig. 2.12 (column 3) show examples where NOC describes a large variety of objects from ImageNet. Fig. 2.10 compares our model with DCC. Fig. 2.11 and Fig. 2.12 (right) outline some errors. Failing to describe a new object is one common error for NOC. E.g. Fig. 2.12 (top right), NOC incorrectly describes a man holding a "sitar" as a man holding a "baseball bat". Other common errors include generating non-grammatical or nonsensical phrases (example with "gladiator", "aardvark") and repeating a specific object ("A barracuda ... with a barracuda", "trifle cake").

2.6.3.2 Describing Rare Objects/Words

The selected rare words occur with varying frequency in the MSCOCO training set, with about 52 mentions on average (median 27) across all training sentences. For example, words such as "bonsai" only appear 5 times,"whisk" (11 annotations), "teapot" (30 annotations), and others such as pumpkin appears 58 times, "swan" (60 annotations), and on the higher side objects like scarf appear 144 times. When tested on ImageNet images containing these concepts, a model trained only with MSCOCO paired data incorporates rare words into sentences 2.93% of the time with an average F1 score of 4.58%. In contrast, integrating outside data, our NOC model can incorporate rare words into descriptions 35.15% of the time with an average F1 score of 47.58%. We do not

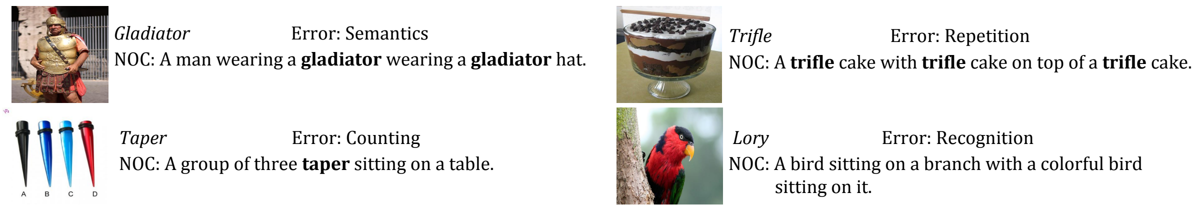

Gladiator
Error: Semantics
NOC: A man wearing a gladiator wearing a gladiator hat.
Trifle
Error: Repetition
NOC: A trifle cake with trifle cake on top of a trifle cake.
Taper
Error: Counting
NOC: A group of three taper sitting on a table.
A B C D
Lory
Error: Recognition
NOC: A bird sitting on a branch with a colorful bird sitting on it.

Figure 2.11: ImageNet Captioning: Common types of errors observed in the captions generated by the NOC model.

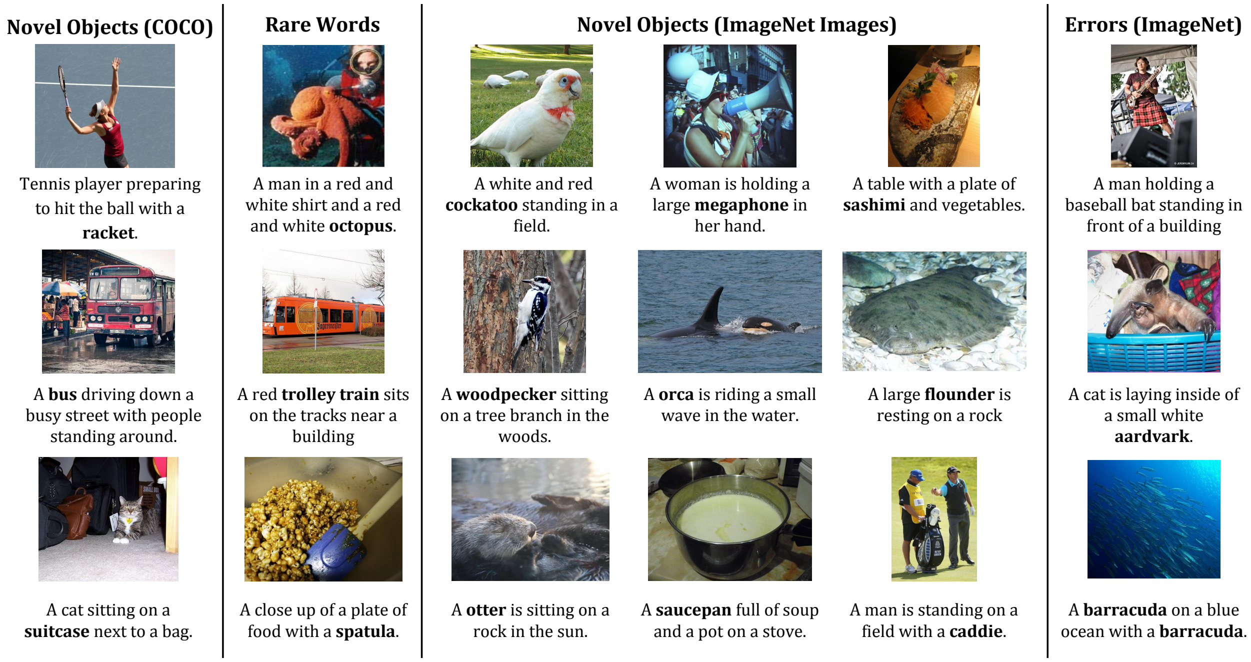

Novel Objects (COCO)
Rare Words
Novel Objects (ImageNet Images)
Errors (ImageNet)
Tennis player preparing to hit the ball with a racket.
A man in a red and white shirt and a red and white octopus.
A white and red cockatoo standing in a field.
A woman is holding a large megaphone in her hand.
A table with a plate of sashimi and vegetables.
A man holding a baseball bat standing in front of a building
A bus driving down a busy street with people standing around.
A red trolley train sits on the tracks near a building
A woodpecker sitting on a tree branch in the woods.
A orca is riding a small wave in the water.
A large flounder is resting on a rock
A cat is laying inside of a small white aardvark.
A cat sitting on a suitcase next to a bag.
A close up of a plate of food with a spatula.
A otter is sitting on a rock in the sun.
A saucepan full of soup and a pot on a stove.
A man is standing on a field with a caddie.
A barracuda on a blue ocean with a barracuda.

Figure 2.12: Descriptions produced by NOC on a variety of objects, including “caddie”, “saucepan”, and “flounder”. (Right) NOC makes errors and (top right) fails to describe the new object (“sitar”). More categories of images and objects are in the supplement.

compare this to DCC since DCC cannot be applied directly to caption rare objects.

2.6.3.3 Human Evaluation

ImageNet images do not have accompanying captions and this makes the task much more challenging to evaluate. To compare the performance of NOC and DCC we obtain human judgements on captions generated by the models on several object categories. We select 3 images each from about 580 object categories that at least one of the two models, NOC and DCC, can describe. (Note that although both models were trained on the same ImageNet object categories, NOC is able to describe almost all of the object categories that have been described by DCC). When selecting the

	Word Incorporation		Image Description	
Objects subset →	Union	Intersection	Union	Intersection
NOC is better	**43.78**	34.61	**59.84**	51.04
DCC is better	25.74	34.12	40.16	48.96
Both equally good	6.10	9.35	-	
Neither is good	24.37	21.91	-	

Table 2.10: ImageNet: Human judgements comparing our NOC model with DCC [Hen+16a] on the ability to meaningfully incorporate the novel object in the description (Word Incorporation) and describe the image. 'Union' and 'Intersection' refer to the subset of objects where at least one model, and both models are able to incorporate the object name in the description. All values in %.

images, for object categories that both models can describe, we make sure to select at least two images for which both models mention the object label in the description. Each image is presented to three workers. We conducted two human studies: Given the image, the ground-truth object category (and meaning), and the captions generated by the models, we evaluate on:

Word Incorporation: We ask humans to choose which sentence/caption incorporates the object label meaningfully in the description. The options provided are: (i) Sentence 1 incorporates the word better, (ii) Sentence 2 incorporates the word better, (iii) Both sentences incorporate the word equally well, or (iv) Neither of them do well.

Image Description: We also ask humans to pick which of the two sentences describes the image better.

This allows us to compare both how well a model incorporates the novel object label in the sentence, as well as how appropriate the description is to the image. The results are presented in Table 2.10. On the subset of images corresponding to objects that both models can describe (Intersection), NOC and DCC appear evenly matched, with NOC only having a slight edge. However, looking at all object categories (Union), NOC is able to both incorporate the object label in the sentence, and describe the image better than DCC.

2.7 Discussion

Our work on novel object captioning stems from a rich body of research on deep captioning, zero-shot learning, and other work which studies describing new objects in context. After publication of our work, a variety of other researchers have considered the problem of novel object captioning. In this section we consider both prior work, and work that has stemmed from our initial contribution.

2.7.1 Related Work.

Deep Captioning A variety of early deep description models [Don+15; Vin+15; KFF15; KSZ14; Fan+15; Mao+15a] achieved promising results on the image captioning task. Some [Don+15; Vin+15; KFF15] follow a CNN-RNN framework: first high-level features are extracted from a CNN trained on the image classification task, and then a recurrent model learns to predict subsequent words of a caption conditioned on image features and previously predicted words. Others [KSZ14; Mao+15a] adopt a multimodal framework in which recurrent language features and image features are embedded in a multimodal space. The multimodal embedding is then used to predict the caption word by word. Retrieval methods [Dev+15] based on comparing the k-nearest neighbors of training and test images in a deep image feature space, have also achieved competitive results on the captioning task. However, retrieval methods are limited to words and descriptions which appear in a training set of paired image-sentence data. As opposed to using high level image features extracted from a CNN, another approach [Fan+15; Wu+16] is to train classifiers on visual concepts such as objects, attributes and scenes. A language model, such as an LSTM [Wu+16] or maximum entropy model [Fan+15], then generates a visual description conditioned on the presence of classified visual elements. Our models for novel object captioning most closely resemble the framework suggested in [Mao+15a] which uses a multimodal space to combine features from image and language. By fusing languistic and visual information later in the network, we are able to train and transfer information from separately trained language and image models more effectively.

Zero-Shot Learning Zero-shot learning has received substantial attention in computer vision [Roh+10; PG11; LNH14; Soc+13; Fro+13] since it becomes difficult to obtain sufficient labeled images as the number of object categories grows. In particular, we draw on previous zero-shot learning work that mines object relationships from external text data [Roh+10; Soc+13; Fro+13]. [Roh+10] uses text corpora to determine how objects are related to each other, then classifies unknown objects based on their relationship to known objects. In [Soc+13; Fro+13], images are mapped to semantic word vectors corresponding to their classes, and the resulting image embeddings are used to detect and distinguish between unseen and seen classes. We also exploit transfer learning via an intermediate-level semantic word vector representation, however, the above approaches focus specifically on assigning a category label, while our method generates full sentence descriptions.

Describing New Objects in Context Many early caption models [Tho+14; Kri+13; Gua+13; Kul+13; Gup+09] rely on first discerning visual elements from an image, such as subjects, objects, scenes, and actions, then filling in a sentence template to create a coherent visual description. These models are capable of describing objects without being provided with paired image-sentence examples containing the objects, but are restricted to generating descriptions using a fixed, predetermined template. [Mao+15b] explore describing new objects with a deep caption model with only a few paired image-sentence examples during training. However, [Mao+15b] do not consider how to describe objects when no paired image-sentence data is available.

2.7.2 Concurrent and Future Work

Since introducing the task of novel object captioning, a variety of other novel object captioning frameworks have been proposed [And+17; Yao+17; Lu+18; Wu+18; AGJ18]. One drawback of NOC and DCC is that they require changing the underlying architecture of the description system. Thus, as researchers build systems which are generally better at the captioning task, these contributions cannot be immediately integrated into the DCC or NOC models. Methods like constrained beamsearch [And+17], which proposes an inference based beamsearch method for describing novel objects, or partially supervised image captioning [AGJ18], which proposes a loss for describing novel objects, are agnostic to the underlying caption architecture. Thus, such systems can continue to achieve improved results as captioning systems improve. Other recent work [Lu+18; Wu+18] integrate detection frameworks (as opposed to classification frameworks) into novel object captioning architectures. Integrating detection leads to stronger visual grounding and improved performance.

Since our initial work, others have also proposed new datasets and testing protocols to study description models that are compositional. Recently, the nocaps [Agr+18] dataset was proposed to supply a large-scale dataset for researchers to test their models on novel object captioning at scale. [Lu+18] propose a new split of MSCOCO called the *robust* split which splits the MSCOCO dataset such that certain pairs of objects (e.g., “chair” and “cat”) are seen individually at train time, but are only seen together in an image at test time. Such new datasets and testing protocols are likely to encourage continued research on caption models which are robust to novel objects and new compositions of objects during inference.

Though we have made great progress in the field of image captioning, there is still considerable room to build models which can truly caption images at scale. Though considerable work has focussed on novel object captioning, less work has explicitly considered rare object captioning, as we did when studying our NOC model. Furthermore, even when examples of objects are present at training time, captioning systems can be driven to generate sentences based on linguistic bias, as opposed to visual understanding of a scene. For example, if more dogs appear on couches at training time than on beds, a caption model might be driven to over predict dogs on couches at inference time. In the next chapter, we more closely consider how language bias interferes with image captioning. We then propose a solution for a special, yet important case, of bias in visual description – gender bias.

Chapter 3

Object Hallucination in Image Captioning

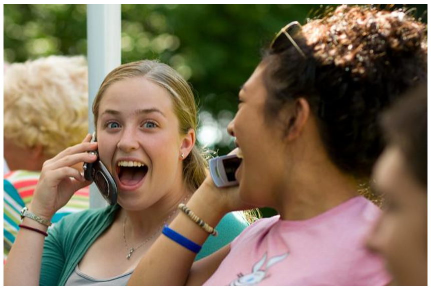
3.1 Introduction

In Figure 3.1 we show an example where a competitive captioning model, Neural Baby Talk (NBT) [Lu+18], hallucinates the object "bench." [1] Describing objects that are *not present* in an image has been shown to be undesirable to humans. For example, the LSMDC challenge [Roh+17b] documents that correctness is more important to human judges than specificity. In another study, [Mac+17] analyzed how visually impaired people react to automatic image captions. They found that people vary in their preference of either coverage or correctness. For many visually impaired who value correctness over coverage, hallucination is an obvious concern. Besides being poorly received by humans, object hallucination reveals an internal issue of a captioning model, such as relying on learned biases.

Here, we assess the phenomenon of object hallucination in contemporary captioning models, and consider several key questions. The first question we aim to answer is: *Which models are more prone to hallucination?* We analyze this question on a diverse set of captioning models, spanning different architectures and learning objectives. To measure object hallucination, we propose a new metric, *CHAIR (Caption Hallucination Assessment with Image Relevance)*, which captures image relevance of the generated captions. Specifically, we consider both ground truth object annotations (MSCOCO Object segmentation [Lin+14b]) and ground truth sentence annotations (MSCOCO Captions [Che+15]). Interestingly, we find that models which score best on standard sentence metrics do not always hallucinate less.

The second question we raise is: *What are the likely causes of hallucination?* While hallucination may occur due to a number of reasons, we believe the top factors include visual misclassification and over-reliance on language priors. The latter may result in memorizing which words "go together" regardless of image content, which may lead to poor generalization, once the test distribution is changed. We propose *image and language model consistency* scores to investigate this issue, and find that models which hallucinate more tend to make mistakes consistent with a

[1] This chapter is based on joint work done with Anna Rohrbach, Kaylee Burns, Trevor Darrell, and Kate Saenko [Roh+18] presented at EMNLP 2018.



NBT: A woman talking on a cell phone while sitting on a ***bench***.
CIDEr: **0.87**, METEOR: 0.23, SPICE: **0.22,** CHs: **1.00**, CHi: **0.33**

TopDown: A woman is talking on a cell phone.
CIDEr: 0.54, METEOR: **0.26**, SPICE: 0.13, CHs: **0.00**, CHi: **0.00**

Figure 3.1: Image captioning models often "hallucinate" objects that may appear in a given context, like e.g. a *bench* here. Moreover, the sentence metrics do not always appropriately penalize such hallucination. Our proposed metrics (CHAIRs and CHAIRi) reflect hallucination. For CHAIR *lower is better*.

language model.

Finally, we ask: *How well do the standard metrics capture hallucination?* It is a common practice to rely on automatic sentence metrics, e.g. CIDEr [VLZP15], to evaluate captioning performance during development, and few employ human evaluation to measure the final performance of their models. As we largely rely on these metrics, it is important to understand how well they capture the hallucination phenomenon. In Figure 3.1 we show how two sentences, from NBT with hallucination and from TopDown model [And+18a] – without, are scored by the standard metrics. As we see, hallucination is not always appropriately penalized. We find that by using additional ground truth data about the image in the form of object labels, our metric CHAIR allows us to catch discrepancies that the standard captioning metrics cannot fully capture. We then investigate ways to assess object hallucination risk with the standard metrics. Finally, we show that CHAIR is complementary to the standard metrics in terms of capturing human preference.

3.2 Caption Hallucination Assessment

We first introduce our image relevance metric, *CHAIR*, which assesses captions w.r.t. objects that are actually in an image. It is used as a main tool in our evaluation. Next we discuss the notions of *image and language model consistency*, which we use to reason about the causes of hallucination.

3.2.1 The CHAIR Metric

To measure object hallucination, we propose the *CHAIR* metric, which calculates what proportion of words generated are actually in the image according to the ground truth sentences and object segmentations. This metric has two variants: per-instance, or what fraction of object instances are hallucinated (denoted as CHAIRi), and per-sentence, or what fraction of sentences include a hallucinated object (denoted as CHAIRs):

$$\text{CHAIR}_i = \frac{|\{\text{hallucinated objects}\}|}{|\{\text{all objects mentioned}\}|}$$

$$\text{CHAIR}_s = \frac{|\{\text{sentences with hallucinated object}\}|}{|\{\text{ all sentences}\}|}$$

For easier analysis, we restrict our study to the 80 MSCOCO objects which appear in the MSCOCO segmentation challenge. To determine whether a generated sentence contains hallucinated objects, we first tokenize each sentence and then singularize each word. We then use a list of synonyms for MSCOCO objects (based on the list from [Lu+18]) to map words (e.g., "player") to MSCOCO objects (e.g., "person"). Additionally, for sentences which include two word compounds (e.g., "hot dog") we take care that other MSCOCO objects (in this case "dog") are not incorrectly assigned to the list of MSCOCO objects in the sentence. For each ground truth sentence, we determine a list of MSCOCO objects in the same way. The MSCOCO segmentation annotations are used by simply relying on the provided object labels.

We find that considering both sources of annotation is important. For example, MSCOCO contains an object "dining table" annotated with segmentation maps. However, humans refer to many different kinds of objects as "table" (e.g., "coffee table" or "side table"), though these objects are not annotated as they are not specifically "dining table". By using sentence annotations to scrape ground truth objects, we account for variation in how human annotators refer to different objects. Inversely, we find that frequently humans will not mention all objects in a scene. Qualitatively, we observe that both annotations are important to capture hallucination. Empirically, we verify that using only segmentation labels or only reference captions leads to higher hallucination (and practically incorrect) rates.

3.2.2 Image Consistency

We define a notion of *image consistency*, or how consistent errors from the captioning model are with a model which predicts objects based on an image alone. To measure image consistency for a particular generated word, we train an image model and record $P(w|I)$ or the probability of predicting the word given only the image. To score the image consistency of a caption we use the average of $P(w|I)$ for all MSCOCO objects, where higher values mean that errors are *more* consistent with the image model. Our image model is a multi-label classification model with labels corresponding to MSCOCO objects (labels determined the same way as is done for CHAIR) which shares the visual features with the caption models.

3.2.3 Language Consistency

We also introduce a notion of *language consistency*, i.e. how consistent errors from the captioning model are with a model which predicts words based only on previously generated words. We train

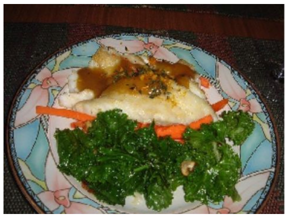


Image Model predictions:
bowl, broccoli, carrot, dining table

Language Model predictions for the last word:
fork, spoon, bowl

Generated caption: A plate of food with broccoli and a ***fork***.

Figure 3.2: Example of image and language consistency. The hallucination error ("fork") is more consistent with the Language Model.

an LSTM [HS97] based language model which predicts a word w_t given previous words $w_{0:t-1}$ on MSCOCO data. We report language consistency as $1/R(w_t)$ where $R(w_t)$ is the rank of the predicted word in the language model. Again, for a caption we report average rank across all MSCOCO objects in the sentence and higher language consistency implies that errors are *more* consistent with the language model.

We illustrate image and language consistency in Figure 3.2, i.e. the hallucination error ("fork") is more consistent with the Language Model predictions than with the Image Model predictions. We use these consistency measures in Section 3.3.3 to help us investigate the causes of hallucination.

3.3 Evaluation

In this section we present the findings of our study, where we aim to answer the following questions: *Which models are more prone to hallucination? What are the likely causes of hallucination? How well do the standard metrics capture hallucination?*

3.3.1 Baseline Captioning Models

We compare object hallucination across a wide range of models. We define two axes for comparison: model architecture and learning objective.

Model architecture. Regarding model architecture, we consider models both with and without attention mechanisms. In this work, we use "attention" to refer to any mechanism which learns to focus on different image regions, whether image regions be determined by a high level feature map, or by object proposals from a trained detector. All models are end-to-end trainable and use a recurrent neural network (LSTM [HS97] in our case) to output text. For non-attention based methods we consider the **FC model** from [Ren+17] which incorporates visual information by initializing the LSTM hidden state with high level image features. We also consider **LRCN** [Don+15] which considers visual information at each time step, as opposed to just initializing the LSTM hidden state with extracted features.

		Cross Entropy					Self Critical				
Model	Att.	S	M	C	CHs	CHi	S	M	C	CHs	CHi
LRCN*		17.0	23.9	90.8	17.7	12.6	16.9	23.5	93.0	17.7	12.9
FC*		17.9	24.9	95.8	15.4	11.0	18.4	25.0	103.9	14.4	10.1
Att2In*	✓	18.9	25.8	102.0	10.8	7.9	19.0	25.7	106.7	12.2	8.4
TopDown*	✓	19.9	26.7	107.6	8.4	6.1	20.4	27.0	117.2	13.6	8.8
TopDown-BB †	✓	20.4	27.1	113.7	8.3	5.9	21.4	27.7	120.6	10.4	6.9
NBT †	✓	19.4	26.2	105.1	7.4	5.4	-	-	-	-	
		Cross Entropy					GAN				
GAN ‡		18.7	25.7	100.4	10.7	7.7	16.6	22.7	79.3	8.2	6.5

Table 3.1: Hallucination analysis on the Karpathy Test set: Spice (S), CIDEr (C) and METEOR (M) scores across different image captioning models as well as CHAIRs (sentence level, CHs) and CHAIRi (instance level, CHi). All models are generated with beam search (beam size=5). * are trained/evaluated within the same implementation [Luo+18], † are trained/evaluated with implementation publicly released with corresponding papers, and ‡ sentences obtained directly from the author. For discussion see Section 3.3.2.

For attention based models, we consider **Att2In** [Ren+17], which is similar to the original attention based model proposed by [Xu+15a], except the image feature is only input into the cell gate as this was shown to lead to better performance. We then consider the attention model proposed by [And+18a] which proposes a specific "top-down attention" LSTM as well as a "language" LSTM. Generally attention mechanisms operate over high level convolutional layers. The attention mechanism from [And+18a] can be used on such feature maps, but Anderson et al. also consider feature maps corresponding to object proposals from a detection model. We consider both models, denoted as **TopDown** (feature map extracted from high level convolutional layer) and **TopDown-BB** (feature map extracted from object proposals from a detection model). Finally, we consider the recently proposed **Neural Baby Talk (NBT)** model [Lu+18] which explicitly uses object detections (as opposed to just bounding boxes) for sentence generation.

Learning objective. All of the above models are trained with the standard *cross entropy* (CE) loss as well as the *self-critical* (SC) loss proposed by [Ren+17] (with an exception of NBT, where only the CE version is included). The SC loss directly optimizes the CIDEr metric with a reinforcement learning technique. We additionally consider a model trained with a *GAN* loss [She+17b] (denoted **GAN**), which applies adversarial training to obtain more diverse and "human-like" captions, and their respective non-GAN baseline with the CE loss.

TopDown deconstruction. To better evaluate how each component of a model might influence hallucination, we "deconstruct" the TopDown model by gradually removing components until it is equivalent to the FC model. The intermediate networks are *NoAttention*, in which the attention mechanism is replaced by mean pooling, *NoConv* in which spatial feature maps are not input into the network (the model is provided with fully connected feature maps), *SingleLayer* in which only one LSTM is included in the model, and finally, instead of inputting visual features at each time step, visual features are used to initialize the LSTM embedding as is done in the FC model.

By deconstructing the TopDown model in this way, we ensure that model design choices and hyperparameters do not confound results.

Implementation details. All the baseline models employ features extracted from the fourth layer of ResNet-101 [He+16], except for the GAN model which employs ResNet-152. Models without attention traditionally use fully connected layers as opposed to convolutional layers. However, as ResNet-101 does not have intermediate fully connected layers, it is standard to average pool convolutional activations and input these features into non-attention based description models. Note that this means the difference between the *NoAttention* and *NoConv* model is that the *NoAttention* model learns a visual embedding of spatial feature maps as opposed to relying on pre-pooled feature maps. All models except for TopDown-BB, NBT, and GAN are implemented in the same open source framework from [Luo+18].[2]

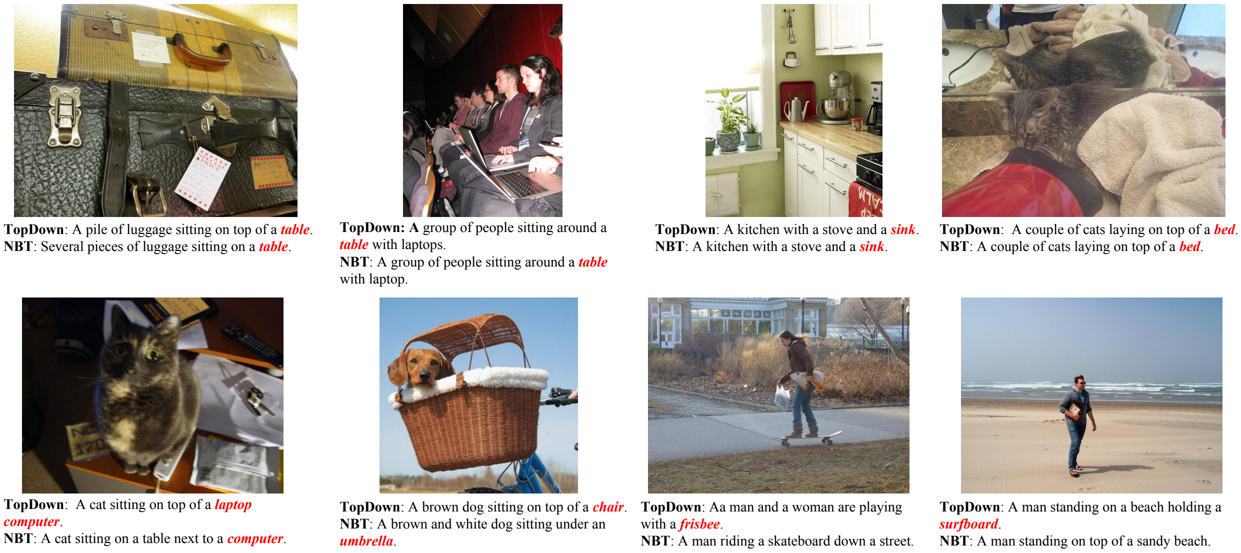
Training/Test splits. We evaluate the captioning models on two MSCOCO splits. First, we consider the split from Karpathy et al. [KFF15], specifically in that case the models are trained on the respective Karpathy Training set, tuned on Karpathy Validation set and the reported numbers are on the Karpathy Test set. We also consider the *Robust* split, introduced in [Lu+18], which provides a compositional split for MSCOCO. Specifically, it is ensured that the object pairs present in the training, validation and test captions do not overlap. In this case the captioning models are trained on the Robust Training set, tuned on the Robust Validation set and the reported numbers are on the Robust Test set.

3.3.2 Which Models Are More Prone To Hallucination?

We first present how well competitive models perform on our proposed CHAIR metric (Table 3.1). We report CHAIR at sentence-level and at instance-level (CHs and CHi in the table). In general, we see that models which perform better on standard evaluation metrics, perform better on CHAIR, though this is not always true. In particular, models which optimize for CIDEr frequently hallucinate more. Out of all generated captions on the Karpathy Test set, anywhere between 7.4% and 17.7% include a hallucinated object. When shifting to more difficult training scenarios in which new combinations of objects are seen at test time, hallucination consistently increases (Table 3.2).

Karpathy Test set. Table 3.1 presents object hallucination on the Karpathy Test set. All sentences are generated using beam search and a beam size of 5. We note a few important trends. First, models with attention tend to perform better on the CHAIR metric than models without attention. As we explore later, this is likely because they have a better understanding of the image. In particular, methods that incorporate bounding box attention (as opposed to relying on coarse feature maps), consistently have lower hallucination as measured by our CHAIR metric. Note that the NBT model does not perform as well on standard captioning metrics as the TopDown-BB model but has lower hallucination. This is perhaps because bounding box proposals come from the MSCOCO detection task and are thus “in-domain” as opposed to the TopDown-BB model which relies on proposals learned from the Visual Genome [Kri+17] dataset. Second, frequently training models with the self-critical loss actually increases the amount of hallucination. One hypothesis

[2] https://github.com/ruotianluo/self-critical.pytorch



TopDown: A pile of luggage sitting on top of a ***table***.
NBT: Several pieces of luggage sitting on a ***table***.

TopDown: A group of people sitting around a ***table*** with laptops.
NBT: A group of people sitting around a ***table*** with laptop.

TopDown: A kitchen with a stove and a ***sink***.
NBT: A kitchen with a stove and a ***sink***.

TopDown: A couple of cats laying on top of a ***bed***.
NBT: A couple of cats laying on top of a ***bed***.

TopDown: A cat sitting on top of a ***laptop computer***.
NBT: A cat sitting on a table next to a ***computer***.

TopDown: A brown dog sitting on top of a ***chair***.
NBT: A brown and white dog sitting under an ***umbrella***.

TopDown: Aa man and a woman are playing with a ***frisbee***.
NBT: A man riding a skateboard down a street.

TopDown: A man standing on a beach holding a ***surfboard***.
NBT: A man standing on top of a sandy beach.

Figure 3.3: Examples of object hallucination from two state-of-the-art captioning models, TopDown and NBT, see Section 3.3.2.

is that CIDEr does not penalize object hallucination sufficiently, leading to both increased CIDEr and increased hallucination. Finally, the LRCN model has a higher hallucination rate than the FC model, indicating that inputting the visual features only at the first step, instead of at every step, leads to more image relevant captions.

We also consider a GAN based model [She+17b] in our analysis. We include a baseline model (trained with CE) as well as a model trained with the GAN loss.[3] Unlike other models, the GAN model uses a stronger visual network (ResNet-152) which could explain the lower hallucination rate for both the baseline and the GAN model. Interestingly, when comparing the baseline and the GAN model (both trained with ResNet-152), standard metrics decrease substantially, even though human evaluations from [She+17b] demonstrate that sentences are of comparable quality. On the other hand, hallucination decreases, implying that the GAN loss actually helps decrease hallucination. Unlike the self critical loss, the GAN loss encourages sentences to be human-like as opposed to optimizing a metric. Human-like sentences are not likely to hallucinate objects, and a hallucinated object is likely a strong signal to the discriminator that a sentence is generated, and is not from a human.

We also assess the effect of beam size on CHAIR. We find that generally beam search decreases hallucination. We use beam size of 5, and for all models trained with cross entropy, it outperforms lower beam sizes on CHAIR. However, when training models with the self-critical loss, beam size sometimes leads to worse performance on CHAIR. For example, on the Att2In model trained with SC loss, a beam size of 5 leads to 12.2 on CHAIRs and 8.4 on CHAIRi, while a beam size of 1 leads to 10.8 on CHAIRs and 8.1 on CHAIRi.

[3] Sentences were procured directly from the authors.

	Att	S	M	C	CHs	CHi
FC*		15.5	22.7	76.2	21.3	15.3
Att2In*	✓	16.9	24.0	85.8	14.1	10.1
TopDown*	✓	17.7	24.7	89.8	11.3	7.9
NBT †	✓	18.2	24.9	93.5	6.2	4.2

Table 3.2: Hallucination Analysis on the Robust Test set: Spice (S), CIDEr (C) and METEOR (M) scores across different image captioning models as well as CHAIRs (sentence level, CHs) and CHAIRi (instance level, CHi). * are trained/evaluated within the same implementation [Luo+18], † are trained/evaluated with implementation publicly released with corresponding papers. All models trained with cross-entropy loss. See Section 3.3.2.

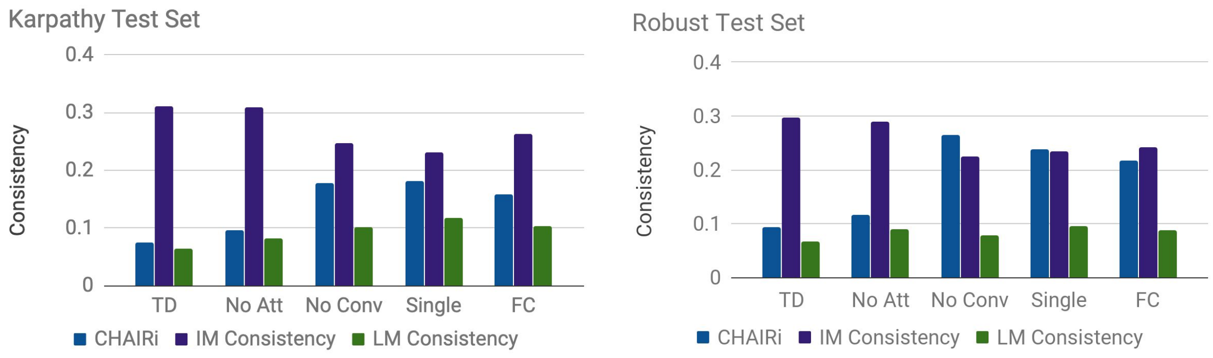


Figure 3.4: Image and Language model consistency (IM, LM) and CHAIRi (instance-level, CHi) on deconstructed TopDown models. Images with less hallucination tend to make errors consistent with the image model, whereas models with more hallucination tend to make errors consistent with the language model, see Section 3.3.3.

Robust Test set. Next we review the hallucination behavior on the Robust Test set (Table 3.2). For almost all models the hallucination increases on the Robust split (e.g. for TopDown from 8.4% to 11.3% of sentences), indicating that the issue of hallucination is more critical in scenarios where test examples can not be assumed to have the same distribution as train examples. We again note that attention is helpful for decreasing hallucination. We note that the NBT model actually has lower hallucination scores on the robust split. This is in part because when generating sentences we use the detector outputs provided by [Lu+18]. Separate detectors on the Karpathy test and robust split are not available and the detector has access to images in the robust split during training. Consequently, the comparison between NBT and other models is not completely fair, but we include the number for completeness.

In addition to the Robust Test set, we also consider a set of MSCOCO in which certain objects are held out, which we call the *Novel Object split* [Hen+16a]. We train on the training set outlined in [Hen+16a] and test on the Karpathy test split, which includes objects unseen during training. Similarly to the Robust Test set, we see hallucination increase substantially on this split. For example, for the TopDown model hallucination increases from 8.4% to 12.1% for CHAIRs and 6.0% to 9.1% for CHAIRi.

We find no obvious correlation between the average length of the generated captions and the hallucination rate. Moreover, vocabulary size does not correlate with hallucination either, i.e. models with *more diverse* descriptions may actually *hallucinate less*. We notice that hallucinated objects tend to be mentioned towards *the end of the sentence* (on average at position 6, with average sentence length 9), suggesting that some of the preceding words may have triggered hallucination. We investigate this below.

Which objects are hallucinated and in what context? Here we analyze which MSCOCO objects tend to be hallucinated more often and what are the common preceding words and image context. Across all models the super-category *Furniture* is hallucinated most often, accounting for $20 - 50\%$ of all hallucinated objects. Other common super-categories are *Outdoor objects*, *Sports* and *Kitchenware*. On the Robust Test set, *Animals* are often hallucinated. The *dining table* is the most frequently hallucinated object across all models (with an exception of GAN, where *person* is the most hallucinated object). We find that often words like "sitting" and "top" precede the "dining table" hallucination, implying the two common scenarios: a person "sitting at the table" and an object "sitting on top of the table" (Figure 3.3, row 1, examples 1, 2). Similar observations can be made for other objects, e.g. word "kitchen" often precedes "sink" hallucination (Figure 3.3, row 1, example 3) and "laying" precedes "bed" (Figure 3.3, row 1, example 4). At the same time, if we look at which objects are actually present in the image (based on MSCOCO object annotations), we can similarly identify that presence of a "cat" co-occurs with hallucinating a "laptop" (Figure 3.3, row 2, example 1), a "dog" – with a "chair" (Figure 3.3, row 2, example 2) etc. In most cases we observe that the hallucinated objects appear in the relevant scenes (e.g. "surfboard" on a beach), but there are cases where objects are hallucinated out of context (e.g. "bed" in the bathroom, Figure 3.3, row 1, example 4).

3.3.3 What Are The Likely Causes Of Hallucination?

In this section we investigate the likely causes of object hallucination. We have earlier described how we deconstruct the TopDown model to enable a controlled experimental setup. We rely on the deconstructed TopDown models to analyze the impact of model components on hallucination.

First, we summarize the hallucination analysis on the deconstructed TopDown models (Table 3.3). Interestingly, the *NoAttention* model does not do substantially worse than the full model (w.r.t. sentence metrics and CHAIR). However, removing Conv input (*NoConv* model) and relying only on FC features, decreases the performance dramatically. This suggests that much of the gain in attention based models is primarily due to *access to feature maps with spatial locality*, not the actual attention mechanism. Also, similar to LRCN vs. FC in Table 3.1, initializing the LSTM hidden state with image features, as opposed to inputting image features at each time step, leads to lower hallucination (*Single Layer* vs. *FC*). This is somewhat surprising, as a model which has access to image information at each time step should be less likely to "forget" image content and

Karpathy Split	S	M	C	CHs	CHi
TD	19.5	26.1	103.4	10.8	7.5
No Attention	18.8	25.6	99.7	14.2	9.5
No Conv	15.7	22.9	81.3	25.7	17.8
Single Layer	15.5	22.7	80.2	25.7	18.2
FC	16.4	23.3	85.1	23.6	15.8

Table 3.3: Hallucination analysis on deconstructed TopDown models with sentence metrics SPICE (S), METEOR (M), and CIDEr (C), CHAIRs (sentence level, CHs) and CHAIRi (instance level, CHi). See Section 3.3.3.

hallucinate objects. However, it is possible that models which include image inputs at each time step with no access to spatial features overfit to the visual features.

Now we investigate what causes hallucination using the deconstructed TopDown models and the *image consistency* and *language consistency* scores, introduced in Sections 3.2.2 and 3.2.3 which capture how consistent the hallucinations errors are with image- / language-only models.

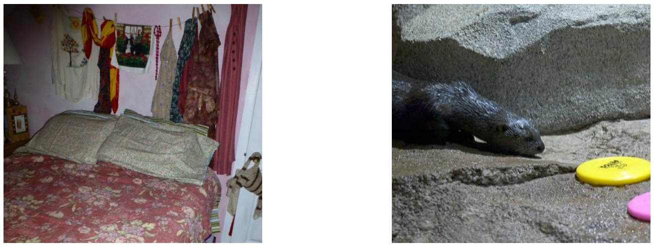
Figure 3.4 shows the CHAIR metric, image consistency and language consistency for the deconstructed TopDown models on the Karpathy Test set (left) and the Robust Test set (right). We note that models with *less* hallucination tend to make errors consistent with the image model, whereas models with *more* hallucination tend to make errors consistent with the language model. This implies that models with less hallucination are better at integrating knowledge from an image into the sentence generation process. When looking at the Robust Test set, Figure 3.4 (right), which is more challenging, as we have shown earlier, we see that image consistency *decreases* when comparing to the same models on the Karpathy split, whereas language consistency is similar across all models trained on the Robust split. This is perhaps because the Robust split contains novel compositions of objects at test time, and all of the models are heavily biased by language.

Finally, we measure image and language consistency during training for the FC model and note that at the beginning of training errors are more consistent with the language model, whereas towards the end of training, errors are more consistent with the image model. This suggests that models first learn to produce fluent language before learning to incorporate visual information.

3.3.4 How Well Do The Standard Metrics Capture Hallucination?

In this section we analyze how well SPICE [And+16a], METEOR [BL05], and CIDEr [VLZP15] capture hallucination. All three metrics do penalize sentences for mentioning incorrect words, either via an F score (METEOR and SPICE) or cosine distance (CIDEr). However, if a caption mentions enough words correctly, it can have a high METEOR, SPICE, or CIDEr score while still hallucinating specific objects.

Our first analysis tool is the TD-Restrict model. This is a modification of the TopDown model, where we enforce that MSCOCO objects which are not present in an image are *not generated* in the caption. We determine which words refer to objects absent in an image following our approach



TD: A cat is sitting on a bed in a room.
S: 12.1 M: 23.8 C: 69.7
TD Restrict: A bed with a blanket and a pillow on it.
S: 23.5 M: 25.4 C: 52.5

TD: A cat laying on the ground with a frisbee.
S: 8.0 M: 13.1 C: 37.0
TD Restrict: A black and white animal laying on the ground.
S: 7.7 M: 15.9 C: 17.4

Figure 3.5: Examples of how TopDown (TD) sentences change when we enforce that objects cannot be hallucinated: SPICE (S), Meteor (M), CIDEr (C), see Section 3.3.4.

	CIDEr	METEOR	SPICE
FC	0.258	0.240	0.318
Att2In	0.228	0.210	0.284
TopDown	0.185	0.168	0.215

Table 3.4: Pearson correlation coefficients between 1-CHs and CIDEr, METEOR, and SPICE scores, see Section 3.3.4.

in Section 3.2.1. We then set the log probability for such words to a very low value. We generate sentences with the TopDown and TD-Restrict model with beam search of size 1, meaning all words produced by both models are the same, until the TopDown model produces a hallucinated word.

We compare which scores are assigned to such captions in Figure 3.5. TD-Restrict generates captions that do not contain hallucinated objects, while TD hallucinates a "cat" in both cases. In Figure 3.5 (left) we see that CIDEr scores the more correct caption much lower. In Figure 3.5 (right), the TopDown model incorrectly calls the animal a "cat." Interestingly, it then correctly identifies the "frisbee," which the TD-Restrict model fails to mention, leading to lower SPICE and CIDEr.

In Table 3.4 we compute Pearson correlation coefficient between individual sentence scores and the *absence* of hallucination, i.e. $1-$CHAIRs; we find that SPICE consistently correlates higher with $1-$CHAIRs. E.g., for the FC model the correlation for SPICE is 0.32, while for METEOR and CIDEr – around 0.25.

We further analyze the metrics in terms of their predictiveness of hallucination risk. Predictiveness means that a certain score should imply a certain percentage of hallucination. Here we show the results for SPICE and the captioning models FC and TopDown. For each model and a score interval (e.g. $10-20$) we compute the percentage of captions *without* hallucination ($1-$CHAIRs). We plot the difference between the percentages from both models (TopDown - FC) in Figure 3.6.

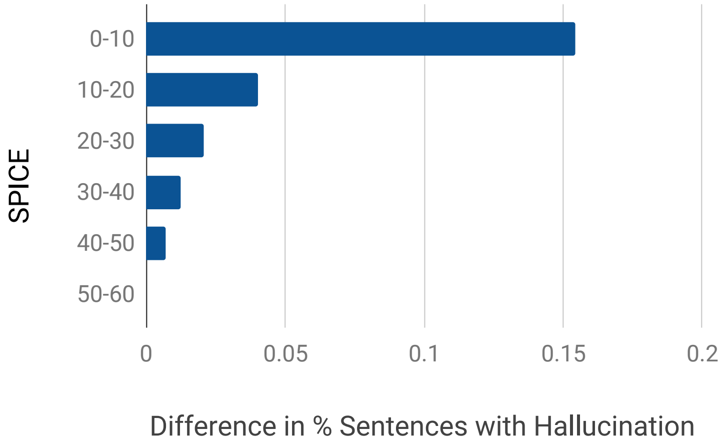


Figure 3.6: Difference in percentage of sentences with *no* hallucination for TopDown and FC models when SPICE scores fall into specific ranges. For sentences with low SPICE scores, the hallucination is generally larger for the FC model, even though the SPICE scores are similar, see Section 3.3.4.

Comparing the models, we note that even when scores are similar (e.g., all sentences with SPICE score in the range of $10-20$), the TopDown model has fewer sentences with hallucinated objects. We see similar trends across other metrics. Consequently, object hallucination can *not* be always predicted based on the traditional sentence metrics.

Is CHAIR complementary to standard metrics? In order to measure usefulness of our proposed metrics, we have conducted the following human evaluation (via the Amazon Mechanical Turk). We have randomly selected 500 test images and respective captions from 5 models: non-GAN baseline, GAN, NBT, TopDown and TopDown - Self Critical. The AMT workers were asked to score the presented captions w.r.t. the given image based on their preference. They could score each caption from 5 (very good) to 1 (very bad). We did not use ranking, i.e. different captions could get the same score; each image was scored by three annotators, and the average score is used as the final human score. For each image we consider the 5 captions from all models and their corresponding sentence scores (METEOR, CIDEr, SPICE). We then compute Pearson correlation between the human scores and sentence scores; we also consider a simple combination of sentence metrics and 1-CHAIRs or 1-CHAIRi by summation. The final correlation is computed by averaging across all 500 images. The results are presented in Table 3.5. Our findings indicate that a simple combination of CHAIRs or CHAIRi with the sentence metrics leads to an increased correlation with the human scores, showing the usefulness and complementarity of our proposed metrics.

Does hallucination impact generation of other words? Hallucinating objects impacts sentence quality not only because an object is predicted incorrectly, but also because the hallucinated word impacts generation of other words in the sentence. Comparing the sentences generated by Top-

	Metric	Metric +(1-CHs)	Metric +(1-CHi)
METEOR	0.269	0.299	0.304
CIDEr	0.282	0.321	0.322
SPICE	0.248	0.277	0.281

Table 3.5: Pearson correlation coefficients between individual/combined metrics and human scores. See Section 3.3.4.

Down and TD-Restrict allows us to analyze this phenomenon. We find that after the hallucinated word is generated, the following words in the sentence are different 47.3% of the time. This implies that hallucination impacts sentence quality beyond simply naming an incorrect object. We observe that one hallucination may lead to another, e.g. hallucinating a "cat" leading to hallucinating a "chair", hallucinating a "dog" – to a "frisbee".

3.4 Discussion

We have started our discussion on bias focussing on one type of prominent error in image captioning models: object hallucination. A significant number of objects are hallucinated in current captioning models (between 5.5% and 13.1% of MSCOCO objects). Furthermore, hallucination is not always captured by the standard captioning metrics. For instance, the popular self critical loss increases CIDEr score, but also the amount of hallucination. Additionally, we find that given two sentences with similar CIDEr, SPICE, or METEOR scores from two different models, the number of hallucinated objects might be quite different. This is especially apparent when standard metrics assign a low score to a generated sentence. Thus, for challenging caption tasks on which standard metrics are currently poor (e.g., the LSMDC dataset [Roh+17a]), the CHAIR metric might be helpful to tease apart the most favorable model. Our results indicate that CHAIR complements the standard sentence metrics in capturing human preference.

Additionally, attention lowers hallucination, but it appears that much of the gain from attention models is due to access to the underlying convolutional features as opposed the attention mechanism itself. Furthermore, we see that models with stronger *image consistency* frequently hallucinate fewer objects, suggesting that strong visual processing is important for avoiding hallucination.

Though our analysis on hallucination can help us better analyze bias, eliminating bias from image captioning models (as well as other machine learning models [RHDV17; RAL18]) is an ongoing research direction. One challenge in mitigating bias is that some biases are beneficial; for example predicting if an object is a "computer mouse" might be easier if a contextual clue, such as a computer, is also in the image. Understanding which learned biases are useful is a difficult and unsolved problem. However, there are some types of predictions which we argue should not be driven by contextual cues; for example, predicting a person's gender. In the next chapter we will move beyond just analyzing and understanding errors cause by learned biases, but mitigating gender bias in image captions using the Equalizer model.

Chapter 4

Mitigating Gender Bias in Image Captioning

4.1 Problem Statement

Exploiting contextual cues can frequently lead to better performance on computer vision tasks [TS01; Tor02; GGM15]. For example, in the visual description task, predicting a "mouse" might be easier given that a computer is also in the image. However, in some cases making decisions based on context can lead to incorrect, and perhaps even offensive, predictions. In this work, we consider one such scenario: generating captions about men and women. We posit that when description models predict gendered words such as "man" or "woman", they should consider visual evidence associated with the described person, and not contextual cues like location (e.g., "kitchen") or other objects in a scene (e.g., "snowboard"). Not only is it important for description systems to avoid egregious errors (e.g., always predicting the word "man" in snowboarding scenes), but it is also important for predictions to be right for the right reason. For example, Figure 4.1 (left) shows a case where prior work predicts the incorrect gender, while our model accurately predicts the gender by considering the correct gender evidence. Figure 4.1 (right) shows an example where both models predict the correct gender, but prior work does not look at the person when describing the image (it is right for the wrong reasons). [1]

Bias in image captioning is particularly challenging to overcome because of the multimodal nature of the task; predicted words are not only influenced by an image, but also biased by the learned language model. Though [Zha+17] studied bias for structured prediction tasks (e.g., semantic role labeling), they did not consider the task of image captioning. Furthermore, the solution proposed in [Zha+17] requires access to the entire test set in order to rebalance gender predictions to reflect the distribution in the training set. Consequently, [Zha+17] relies on the assumption that the distribution of genders is the same at training and test time. We make no such assumptions; we consider a more realistic scenario in which captions are generated for images independent of other

[1] This chapter is based on joint work done with Kaylee Burns, Kate Saenko, Trevor Darrell, and Anna Rohrbach [Hen+18c] presented at ECCV 2018.

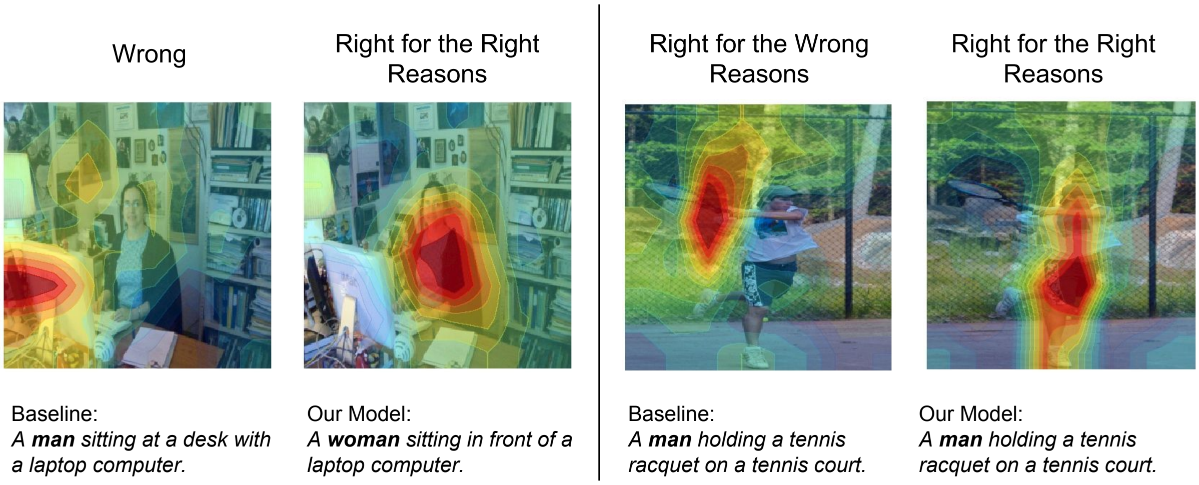

Wrong
Right for the Right Reasons
Right for the Wrong Reasons
Right for the Right Reasons
Baseline:
A **man** sitting at a desk with a laptop computer.
Our Model:
A **woman** sitting in front of a laptop computer.
Baseline:
A **man** holding a tennis racquet on a tennis court.
Our Model:
A **man** holding a tennis racquet on a tennis court.

Figure 4.1: Examples where our proposed model (Equalizer) corrects bias in image captions. The overlaid heatmap indicates which image regions are most important for predicting the gender word. On the left, the baseline predicts gender incorrectly, presumably because it looks at the laptop (not the person). On the right, the baseline predicts the gender correctly but it does not look at the person when predicting gender and is thus not acceptable. In contrast, our model predicts the correct gender word and correctly considers the person when predicting gender.

test images.

In order to encourage description models to generate less biased captions, we introduce the *Equalizer* Model. Our model includes two complementary loss terms: the *Appearance Confusion Loss (ACL)* and the *Confident Loss (Conf)*. The Appearance Confusion Loss is based on the intuition that, given an image in which evidence of gender is absent, description models should be unable to accurately predict a gendered word. However, it is not enough to confuse the model when gender evidence is absent; we must also encourage the model to consider gender evidence when it is present. Our Confident Loss helps to increase the model's confidence when gender is in the image. These complementary losses allow the Equalizer model to be cautious in the absence of gender information and discriminative in its presence.

Our proposed Equalizer model leads to less biased captions: not only does it lead to lower error when predicting gendered words, but it also performs well when the distribution of genders in the test set is not aligned with the training set. Additionally, we observe that Equalizer generates gender neutral words (like "person") when it is not confident of the gender. Furthermore, we demonstrate that Equalizer focuses on humans when predicting gender words, as opposed to focusing on other image context.

4.2 Related Work

Unwanted Dataset Bias. Unwanted dataset biases (e.g., gender, ethnic biases) have been studied across a wide variety of AI domains [RAM18; SC17; Bol+16; Buo17; BS16; PP14]. One common theme is the notion of *bias amplification*, in which bias is not only learned, but amplified [Zha+17; Bol+16; SC17]. For example, in the image captioning scenario, if 70% of images with umbrellas include a woman and 30% include a man, at test time the model might amplify this bias to 85% and 15%. Eliminating bias amplification is not as simple as balancing across attributes for a specific category. [SC17] study bias in classification and find that even though white and black people appear in "basketball" images with similar frequency, models learn to classify images as "basketball" based on the presence of a black person. One explanation is that though the data is balanced in regard to the class "basketball", there are many more white people in the dataset. Consequently, to perfectly balance a dataset, one would have to balance across all possible co-occurrences which is infeasible.

Natural language data is subject to *reporting bias* [Bol+16; GVD13; Mis+16; Mil16] in which people over-report less common co-occurrences, such as "male nurse" [Bol+16] or "green banana" [Mis+16]. [Mil16] also discuss how visual descriptions reflect cultural biases (e.g., assuming a woman with a child is a mother, even though this cannot be confirmed in an image). We observe that annotators specify gender even when gender cannot be confirmed in an image (e.g., a snowboarder might be labeled as "man" even if gender evidence is occluded).

Our work is most similar to [Zha+17] who consider bias in semantic role labeling and multilabel classification (as opposed to image captioning). To avoid bias amplification, [Zha+17] rebalance the test time predictions to more accurately reflect the training time word ratios. This solution is unsatisfactory because (i) it requires access to the entire test set and (ii) it assumes that the distribution of objects at test time is the same as at training time. We consider a more realistic scenario in our experiments, and show that the ratio of woman to man in our predicted sentences closely resembles the ratio in ground truth sentences, even when the test distribution is different from the training distribution.

Fairness. Building AI systems which treat *protected attributes* (e.g., age, gender, sexual orientation) in a fair manner is increasingly important [HPS+16; Dwo+12; ZLM18; QS17]. In the machine learning literature, "fairness" generally requires that systems do not use information such as gender or age in a way that disadvantages one group over another. We consider is different scenario as we are trying to *predict* protected attributes.

Distribution matching has been used to build fair systems [QS17] by encouraging the distribution of decisions to be similar across different protected classes, as well as for other applications such as domain adaption [Tze+15; Zha+15] and transduction learning [QPS09]. Our Appearance Confusion Loss is similar as it encourages the distribution of predictions to be similar for man and woman classes when gender information is not available.

Right for the Right Reasons. Assuring models are "right for the right reasons," or consider similar evidence as humans when making decisions, helps researchers understand how models will perform in real world applications (e.g., when predicting outcomes for pneumonia patients in [Car+15]) or discover underlying dataset bias [Tan+18]. We hypothesize that models which look at appropriate gender evidence will perform better in new scenarios, specifically when the gender distribution at test and training time are different.

Recently, [RHDV17] develop a loss function which compares explanations for a decision to ground truth explanations. However, [RHDV17] generating explanations for visual decisions is a difficult and active area of research [Ram+17; Sel+17; FV17; RSG16; Zin+17; ZF14]. Instead of relying on our model to accurately explain itself during training, we verify that our formulation encourages models to be right for the right reason at test time.

Visual Description. Most visual description work (e.g., [Vin+15; Don+15; KFF15; Xu+15a; And+18a]) focuses on improving overall sentence quality, without regard to captured biases. Though we pay special attention to gender in this work, all captioning models trained on visual description data (MSCOCO [Lin+14b], Flickr30k [You+14], MSR-VTT [Xu+16] to name a few) implicitly learn to classify gender. However current captioning models do not discuss gender the way humans do, but *amplify* gender bias; our intent is to generate descriptions which more accurately reflect human descriptions when discussing this important category.

Gender Classification. Gender classification models frequently focus on facial features [LH15; Zha+16b; EEH14]. In contrast, we are mainly concerned about whether contextual clues in complex scenes bias the production of gendered words during sentence generation. Gender classification has also been studied in natural language processing ([Arg+07; YY06], [Bur+11]).

Ethical Considerations. Frequently, gender classification is seen as a binary task: data points are labeled as either “man” or “woman”. However, AI practitioners, both in industrial[2] and academic[3] settings, are increasingly concerned that gender classification systems should be inclusive. Our captioning model predicts three gender categories: male, female, and gender neutral (e.g., person) based on visual appearance. When designing gender classification systems, it is important to understand where labels are sourced from [Lar17]. We determine gender labels using a previously collected publicly released dataset in which annotators describe images [Lin+14b]. Importantly, people in the images are not asked to identify their gender. Thus, we emphasize that we are not classifying biological sex or gender identity, but rather outward gender appearance.

4.3 Equalizer: Overcoming Bias in Description Models

Equalizer is based on the following intuitions: if evidence to support a specific gender decision is not present in an image, the model should be *confused* about which gender to predict (enforced by an Appearance Confusion Loss term), and if evidence to support a gender decision is in an

[2] https://clarifai.com/blog/socially-responsible-pixels-a-look-inside-clarifais-new-demographics-recognition-model

[3] https://www.media.mit.edu/projects/gender-shades/faq

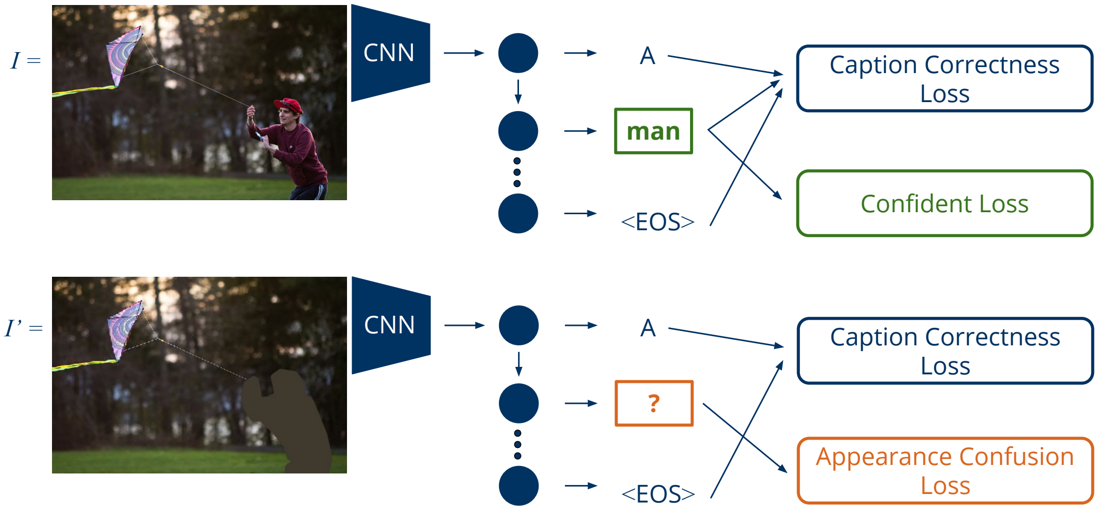


Figure 4.2: Equalizer includes two novel loss terms: the Appearance Confusion Loss on images with men or women (top) and the Confident Loss on images where men and women are occluded (bottom). Together these losses encourage our model to make correct predictions when evidence of gender is present, and be cautious in its absence. We also include the Caption Correctness Loss (cross entropy loss) for both image types.

image, the model should be *confident* in its prediction (enforced by a Confident Loss term). To train our model we require not only pairs of images, I, and sentences, S, but also annotation masks M which indicate which evidence in an image is appropriate for determining gender. Though we use [Vin+15] as our base network, Equalizer is general and can be integrated into any deep description frameworks.

4.3.1 Background: Description Framework

To generate a description, high level image features are first extracted from the InceptionV3 [Sze+16] model. The image features are then used to initialize an LSTM hidden state. To begin sentence generation, a start of sentence token is input into the LSTM. For each subsequent time step during training, the ground truth word w_t is input into the LSTM. At test time, the previously predicted word w_{t-1} is input into the LSTM at each time step. Generation concludes when an end of sequence token is generated. Like [Vin+15], we include the standard cross entropy loss ($\mathcal{L}^{CE}$) during training:

$$\mathcal{L}^{CE} = -\frac{1}{N}\sum_{n=0}^{N}\sum_{t=0}^{T}\log(p(w_t|w_{0:t-1}, I)), \tag{4.1}$$

where N is the batch size, T is the number of words in the sentence, w_t is a ground truth word at time t, and I is an image.

4.3.2 Appearance Confusion Loss

Our Appearance Confusion Loss encourages the underlying description model to be *confused* when making gender decisions if the input image does not contain appropriate evidence for the decision. To optimize the Appearance Confusion Loss, we require ground truth rationales indicating which evidence is appropriate for a particular gender decision. We expect the resulting rationales to be masks, M, which are 1 for pixels which should not contribute to a gender decision and 0 for pixels which are appropriate to consider when determining gender. The Hadamard product of the mask and the original image, $I \odot M$, yields a new image, I', with gender information that the implementer deems appropriate for classification removed. Intuitively, for an image devoid of gender information, the probability of predicting man or woman should be equal. The Appearance Confusion Loss enforces a fair prior by asserting that this is the case.

To define our Appearance Confusion Loss, we first define a *confusion* function ($\mathcal{C}$) which operates over the predicted distribution of words $p(\tilde{w}_t)$, a set of woman gender words ($\mathcal{G}_w$), and a set of man gender words ($\mathcal{G}_m$):

$$\mathcal{C}(\tilde{w}_t, I') = |\sum_{g_w \in \mathcal{G}_w} p(\tilde{w}_t = g_w | w_{0:t-1}, I') - \sum_{g_m \in \mathcal{G}_m} p(\tilde{w}_t = g_m | w_{0:t-1}, I')|. \tag{4.2}$$

In practice, the $\mathcal{G}_w$ consists only of the word "woman" and, likewise, the $\mathcal{G}_m$ consists only of the word "man". These are by far the most commonly used gender words in the datasets we consider and we find that using these "sets" results in similar performance as using more complete sets.

We can now define our Appearance Confusion Loss ($\mathcal{L}^{AC}$) as:

$$\mathcal{L}^{AC} = \frac{1}{N} \sum_{n=0}^{N} \sum_{t=0}^{T} \mathbb{1}(w_t \in \mathcal{G}_w \cup \mathcal{G}_m) \mathcal{C}(\tilde{w}_t, I'), \tag{4.3}$$

where $\mathbb{1}$ is an indicator variable that denotes whether or not w_t is a gendered word.

For the remaining non-gendered words that correspond to images I', we apply the standard cross entropy loss to encourage the model to discuss objects which are still visible in I'. In addition to encouraging sentences to be image relevant even when the gender information has been removed, this also encourages the model to learn representations of words like "dog" and "frisbee" that are not reliant on gender information.

4.3.3 Confident Loss

In addition to being unsure when gender evidence is occluded, we also encourage our model to be confident when gender evidence is present. Thus, we introduce the Confident Loss term, which encourages the model to predict gender words correctly.

Our Confident Loss encourages the probabilities for predicted gender words to be high on images I in which gender information is present. Given functions $\mathcal{F}^W$ and $\mathcal{F}^M$ which measure how confidently the model predicts woman and man words respectively, we can write the Confident Loss as:

$$\mathcal{L}^{Con} = \frac{1}{N}\sum_{n=0}^{N}\sum_{t=0}^{T}(\mathbb{1}(w_t \in \mathcal{G}_w)\mathcal{F}^W(\tilde{w_t}, I) + \mathbb{1}(w_t \in \mathcal{G}_m)\mathcal{F}^M(\tilde{w_t}, I)). \tag{4.4}$$

To measure the confidence of predicted gender words, we consider the quotient between predicted probabilities for man and gender words ($\mathcal{F}^M$ is of the same form):

$$\mathcal{F}^W(\tilde{w_t}, I) = \frac{\sum_{g_m \in \mathcal{G}_m} p(\tilde{w_t} = g_m | w_{0:t-1}, I)}{(\sum_{g_w \in \mathcal{G}_w} p(\tilde{w_t} = g_w | w_{0:t-1}, I)) + \epsilon} \tag{4.5}$$

where ϵ is a small epsilon value added for numerical stability.

When the model is confident of a gender prediction (e.g., for the word “woman”), the probability of the word “woman” should be considerably higher than the probability of the word “man”, which will result in a small value for $\mathcal{F}^W$ and thus a small loss. One nice property of considering the quotient between predicted probabilities is that we encourage the model to distinguish between gendered words without forcing the model to predict a gendered word. For example, if the model predicts a probability of 0.2 for “man”, 0.5 for “woman”, and 0.3 for “person” on a “woman” image, our confidence loss will be low. However, the model is still able to predict gender neutral words, like “person” with relatively high probability. This is distinct from other possible losses, like placing a larger weight on gender words in the cross entropy loss, which forces the model to predict “man”/“woman” words and penalizes the gender neutral words.

4.3.4 The Equalizer Model

Our final model is a linear combination of all aforementioned losses:

$$\mathcal{L} = \alpha\mathcal{L}^{CE} + \beta\mathcal{L}^{AC} + \mu\mathcal{L}^{Con}, \tag{4.6}$$

where α, β, and μ are hyperparameters chosen on a validation set ($\alpha, \mu = 1$, $\beta = 10$ in our experiments).

Our Equalizer method is general and our base captioning framework can be substituted with any other deep captioning framework. By combining all of these terms, the Equalizer model can not only generate image relevant sentences, but also make confident gender predictions under sufficient evidence. We find that both the Appearance Confusion Loss and the Confident Loss are important in creating a confident yet cautious model. Interestingly, the Equalizer model achieves the lowest misclassification rate only when these two losses are combined, highlighting the complementary nature of these two loss terms.

4.4 Experiments

4.4.1 Datasets

MSCOCO-Bias. To evaluate our method, we consider the dataset used by [Zha+17] for evaluating bias amplification in structured prediction problems. This dataset consists of images from MSCOCO [Lin+14b] which are labeled as "man" or "woman". Though "person" is an MSCOCO class, "man" and "woman" are not, so [Zha+17] employ ground truth captions to determine if images contain a man or a woman. Images are labeled as "man" if at least one description includes the word "man" and no descriptions include the word "woman". Likewise, images are labeled as "woman" if at least one description includes the word "woman" and no descriptions include the word "man". Images are discarded if both "man" and "woman" are mentioned. We refer to this dataset as MSCOCO-Bias.

MSCOCO-Balanced. We also evaluate on a set where we purposely change the gender ratio. We believe this is representative of real world scenarios in which different distributions of men and women might be present at test time. The MSCOCO-Bias set has a roughly 1:3 woman to man ratio where as this set, called MSCOCO-Balanced, has a 1:1 woman to man ratio. We randomly select 500 images from MSCOCO-Bias set which include the word "woman" and 500 which include "man".

Person Masks. To train Equalizer, we need ground truth human rationales for why a person should be predicted as a man or a woman. We use the person segmentation masks from the MSCOCO dataset. Once the masked image is created, we fill the segmentation mask with the average pixel value in the image. We use the masks both at training time to compute Appearance Confusion Loss and during evaluation to ensure that models are predicting gender words by looking at the person. While for MSCOCO the person annotations are readily available, for other datasets e.g. a person detector could be used.

4.4.2 Metrics

To evaluate our methods, we rely on the following metrics.

Error. Due to the sensitive nature of prediction for protected classes (gender words in our scenario), we emphasize the importance of a low error. The error rate is the number of man/woman misclassifications, while gender neutral terms are not considered errors. We expect that the best model would rather predict gender neutral words in cases where gender is not obvious.

Gender Ratio. Second, we consider the ratio of sentences which belong to a "woman" set to sentences which belong to a "man" set. We consider a sentence to fall in a "woman" set if it predicts any word from a precompiled list of female gendered words, and respectively fall in a "man" set if it predicts any word from a precompiled list of male gendered words.

Right for the Right Reasons. Finally, to measure if a model is "right for the right reasons" we consider the pointing game [Zha+16a] evaluation. We first create visual explanations for “woman”/“man” using the Grad-CAM approach [Sel+17] as well as saliency maps created by occluding image regions in a sliding window fashion. To measure if our models are right for the right reason, we verify whether the point with the highest activation in the explanation heat map falls in the person segmentation mask.

4.4.3 Training Details

All models are initialized from the Show and Tell model [Vin+15] pre-trained on all of MSCOCO for 1 million iterations (without fine-tuning through the visual representation). Models are trained for additional 500,000 iterations on the MSCOCO-Bias set, fine-tuning through the visual representation (Inception v3 [Sze+16]) for 500,000 iterations.

4.4.4 Baselines and Ablations

Baseline-FT. The simplest baseline is fine-tuning the Show and Tell model through the LSTM and convolutional networks using the standard cross-entropy loss on our target dataset, the MSCOCO-Bias dataset.

Balanced. We train a Balanced baseline in which we re-balance the data distribution at training time to account for the larger number of men instances in the training data. Even though we cannot know the correct distribution of our data at test time, we can enforce our belief that predicting a woman or man should be equally likely. At training time, we re-sample the images of women so that the number of training examples of women is the same as the number of training examples of men.

UpWeight. We also experiment with upweighting the loss value for gender words in the standard cross entropy loss to increase the penalty for a misclassification. For each time step where the ground truth caption says the word “man” or “woman”, we multiply that term in the loss by a constant value (10 in reported experiments). Intuitively, upweighting should encourage the models to accurately predict gender words. However, unlike our Confident Loss, upweighting drives the model to make either “man” or “woman” predictions without the opportunity to place a high probability on gender neutral words.

Ablations. To isolate the impact of the two loss terms in Equalizer, we report results with only the Appearance Confusion Loss (Equalizer w/o Conf) and only the Confidence Loss (Equalizer w/o ACL). We then report results of our full Equalizer model.

4.4.5 Results

Error. Table 4.1 reports the error rates when describing men and women on the MSCOCO-Bias and MSCOCO-Balanced test sets. Comparing to baselines, Equalizer shows consistent improve-

	MSCOCO-Bias		MSCOCO-Balanced	
Model	Error	Ratio Δ	Error	Ratio Δ
Baseline-FT	12.83	0.15	19.30	0.51
Balanced	12.85	0.14	18.30	0.47
UpWeight	13.56	0.08	16.30	0.35
Equalizer w/o ACL	7.57	0.04	10.10	0.26
Equalizer w/o Conf	9.62	0.09	13.90	0.40
Equalizer	**7.02**	**-0.03**	**8.10**	**0.13**

Table 4.1: Evaluation of predicted gender words based on error rate and ratio of generated sentences which include the "woman" words to sentences which include the "man" words. Equalizer achieves the lowest error rate and predicts sentences with a gender ratio most similar to the corresponding ground truth captions (Ratio Δ), even when the test set has a different distribution of gender words than the training set, as is the case for the MSCOCO-Balanced dataset.

ments. Importantly, our full model consistently improves upon Equalizer w/o ACL and Equalizer w/o Conf. When comparing Equalizer to baselines, we see a larger performance gain on the MSCOCO-Balanced dataset. As discussed later, this is in part because our model does a particularly good job of decreasing error on the minority class (woman). Unlike baseline models, our model has a similar error rate on each set. This indicates that the error rate of our model is not as sensitive to shifts in the gender distribution at test time.

Interestingly, the results of the Baseline-FT model and Balanced model are not substantially different. One possibility is that the co-occurrences across words are not balanced (e.g., if there is gender imbalance specifically for images with "umbrella" just balancing the dataset based on gender word counts is not sufficient to balance the dataset). We emphasize that balancing across all co-occurring words is difficult in large-scale settings with large vocabularies.

Gender Ratio We also consider the ratio of captions which include only female words to captions which include only male words. In Table 4.1 we report the *difference* between the ground truth ratio and the ratio produced by each captioning model. Impressively, Equalizer achieves the closest ratio to ground truth on both datasets. Again, the ACL and Confident losses are complementary and Equalizer has the best overall performance.

Performance for Each Gender. Images with females comprise a much smaller portion of MSCOCO than images with males. Therefore the overall performance across classes (i.e. man, woman) can be misleading because it downplays the errors in the minority class. Additionally, unlike [Zha+17] who consider a classification scenario in which the model is forced to predict a gender, our description models can also discuss gender neutral terms such as "person" or "player". In Table 4.2 for each gender, we report the percentage of sentences in which gender is predicted correctly or incorrectly and when no gender specific word is generated on the MSCOCO-Bias set.

Model	Women Correct	 Incorrect	 Other	Men Correct	 Incorrect	 Other	Outcome Divergence between Genders
Baseline-FT	46.28	34.11	19.61	75.05	4.23	20.72	0.121
Balanced	47.67	33.80	18.54	75.89	4.38	19.72	0.116
UpWeight	**60.59**	29.82	9.58	**87.84**	6.98	5.17	0.078
Equalizer w/o ACL	56.18	16.02	27.81	67.58	**4.15**	28.26	0.031
Equalizer w/o Conf	46.03	24.84	29.13	61.11	3.47	35.42	0.075
Equalizer (Ours)	57.38	**12.99**	29.63	59.02	4.61	36.37	**0.018**

Table 4.2: Accuracy per class for MSCOCO-Bias dataset. Though UpWeight achieves the highest recall for both men and women images, it also has a high error, especially for women. One criterion of a "fair" system is that it has similar outcomes across classes. We measure outcome similarity by computing the Jensen-Shannon divergence between Correct/Incorrect/Other sentences for men and women images (lower is better) and observe that Equalizer performs best on this metric.

Across all models, the error for Men is quite low. However, our model significantly improves the error for the minority class, Women. Interestingly, we observe that Equalizer has a similar recall (Correct), error (Incorrect), and Other rate across both genders. A caption model could be considered more "fair" if, for each gender, the possible outcomes (correct gender mentioned, incorrect gender mentioned, gender neutral) are similar. This resembles the notion of equalized odds in fairness literature [HPS+16], which requires a system to have similar false positive and false negative rates across groups. To formalize this notion of fairness in our captioning systems, we report the outcome type divergence between genders by measuring the Jensen-Shannon [Lin91] divergence between Correct/Incorrect/Other outcomes for Men and Women. Lower divergence indicates that Women and Men classes result in a similar distribution of outcomes, and thus the model can be considered more "fair". Equalizer has the lowest divergence (0.018).

Annotator Confidence. As described above, gender labels are mined from captions provided in the MSCOCO dataset. Each image corresponds to five captions, but not all captions for a single image include a gendered word. Counting the number of sentences which include a gendered word provides a rough estimate of how apparent gender is in an image and how important it is to mention when describing the scene.

To understand how well our model captures the way annotators describe people, instead of labeling images as either "man" or "woman", we label images as "man", "woman", or "gender neutral" based on how many annotators mentioned gender in their description. For a specific threshold value T, we consider an image to belong to the "man" or "woman" class if T or more annotators mention the gender in their description, and "gender neutral" otherwise. We can then measure accuracy over these three classes. Whereas a naive solution which restricts vocabulary to include no gender words would have low error as defined in Table 4.1, it would not capture the way humans use gender words when describing images. Indeed, the MSCOCO training set includes over 200,000 instances of words which describe people. Over half of all words used to

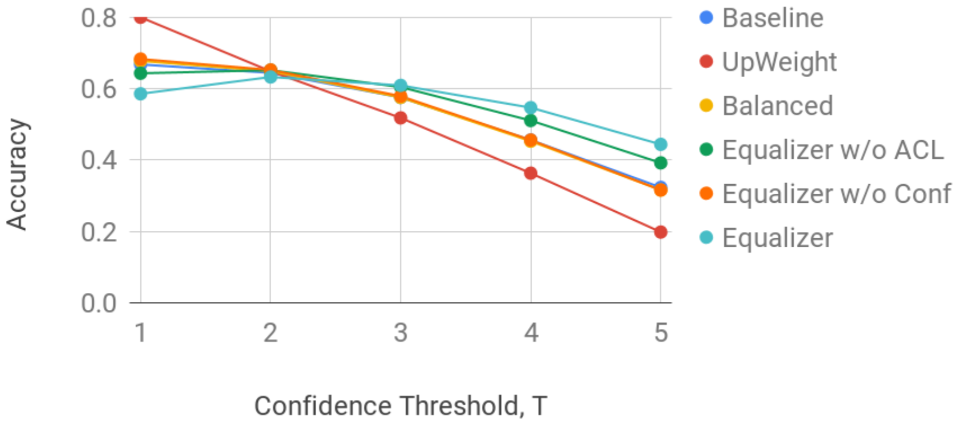


Figure 4.3: Accuracy across man, woman, and gender neutral terms for different models as a function of annotator confidence. When only one annotator describes an image with a gendered word, Equalizer has a low accuracy as it more likely predicts gender neutral words but when more annotations mention gendered words, Equalizer has higher accuracy than other models.

describe people are gendered. By considering accuracy across three classes, we can better measure how well models capture the way humans describe gender.

Figure 4.3 plots the accuracy of each model with respect to the confidence threshold T. At low threshold values, Equalizer performs worse as it tends to more frequently output gender neutral terms, and the UpWeight model, which almost always predicts gendered words, performs best. However, as the threshold value increases, Equalizer performs better than other models, including at a threshold value of 3 which corresponds to classifying images based off the majority vote. This indicates that Equalizer naturally captures when humans describe images with gendered or gender neutral words.

Object Gender Co-Occurrence. We analyze how gender prediction influences prediction of other words on the MSCOCO-Bias test set. Specifically, we consider the 80 MSCOCO categories, excluding the category "person". We adopt the bias amplification metric proposed in [Zha+17], and compute the following ratios: $\frac{count(man\&object)}{count(person\&object)}$ and $\frac{count(woman\&object)}{count(person\&object)}$, where *man* refers to all male words, *woman* refers to all female words, and *person* refers to all male, female, or gender neutral words. Ideally, these ratios should be similar for generated captions and ground truth captions. However, e.g. for *man* and *motorcycle*, the ground truth ratio is 0.40 and for the Baseline-FT and Equalizer, the ratio is 0.81 and 0.65, respectively. Though Equalizer over-predicts this pair, the ratio is closer to the ground truth than when comparing Baseline-FT to the ground truth. Likewise, for *woman* and *umbrella*, the ground truth ratio is 0.40, Baseline-FT ratio is 0.64, and Equalizer ratio is 0.56. As a more holistic metric, we average the *difference* of ratios between ground truth and generated captions across objects (lower is better). For male words, Equalizer is substantially better than the Baseline-FT (0.147 vs. 0.193) and similar for female words (0.096 vs.

Accuracy	Woman	Man	All
Random	22.6	19.5	21.0
Baseline-FT	39.8	34.3	37.0
Balanced	37.6	34.1	35.8
UpWeight	43.3	36.4	39.9
Equalizer w/o ACL	48.1	39.6	43.8
Equalizer w/o Conf	43.9	36.8	40.4
Equalizer (Ours)	**49.9**	**45.2**	**47.5**

(a) Visual explanation is a *Grad-CAM* map.

Accuracy	Woman	Man	All
Random	25.1	17.5	21.3
Baseline-FT	45.3	40.4	42.8
Balanced	48.5	42.2	45.3
UpWeight	54.1	45.5	49.8
Equalizer w/o ACL	54.7	47.5	51.1
Equalizer w/o Conf	48.9	46.7	47.8
Equalizer (Ours)	**56.3**	**51.1**	**53.7**

(b) Visual explanation is a *saliency* map.

Table 4.3: *Pointing game* evaluation that measures whether the visual explanations for "man" / "woman" words fall in the person segmentation ground-truth. Evaluation is done for ground-truth captions on the MSCOCO-Balanced.

0.99).

Caption Quality. Qualitatively, the sentences from all of our models are linguistically fluent (indeed, comparing sentences in Figure 4.4 we note that usually only the word referring to the person changes). However, we do notice a small drop in performance on standard description metrics (25.2 to 24.3 on METEOR [BL05] when comparing Baseline-FT to our full Equalizer) on MSCOCO-Bias. One possibility is that our model is overly cautious and is penalized for producing gender neutral terms for sentences that humans describe with gendered terms.

Right for the Right Reasons. We hypothesize that many misclassification errors occur due to the model looking at the wrong visual evidence, e.g. conditioning gender prediction on context rather than on the person's appearance. We quantitatively confirm this hypothesis and show that our proposed model improves this behavior by looking at the appropriate evidence, i.e. is being "right for the right reasons". To evaluate this we rely on two visual explanation techniques: Grad-CAM [Sel+17] and saliency maps generated by occluding image regions in a sliding window fashion.

Unlike [Sel+17] who apply Grad-CAM to an entire caption, we visualize the evidence for generating specific words, i.e. "man" and "woman". Specifically, we apply Grad-CAM to the last convolutional layer of our image processing network, InceptionV3 [Sze+16], we obtain 8x8 weight matrices. To obtain saliency maps, we resize an input image to 299×299 and uniformly divide it into 32×32 pixel regions, obtaining a 10×10 grid (the bottom/rightmost cells being smaller). Next, for every cell in the grid, we zero out the respective pixels and feed the obtained "partially blocked out" image through the captioning network (similar to as was done in the occlusion sensitivity experiments in [ZF14]). Then, for the ground-truth caption, we compute the "information loss", i.e. the decrease in predicting the words "man" and "woman" as $-\log(p(w_t = g_m))$ and $-\log(p(w_t =$

g_w)), respectively. This is similar to the top-down saliency approach of [Ram+17], who zero-out all the intermediate feature descriptors but one.

To evaluate whether the visual explanation for the predicted word is focused on a person, we rely on person masks, obtained from MSCOCO ground-truth person segmentations. We use the *pointing game* evaluation [Zha+16a]. We upscale visual explanations to the original image size. We define a "hit" to be when the point with the highest weight is contained in the person mask. The accuracy is computed as $\frac{\#hits}{\#hits+\#misses}$.

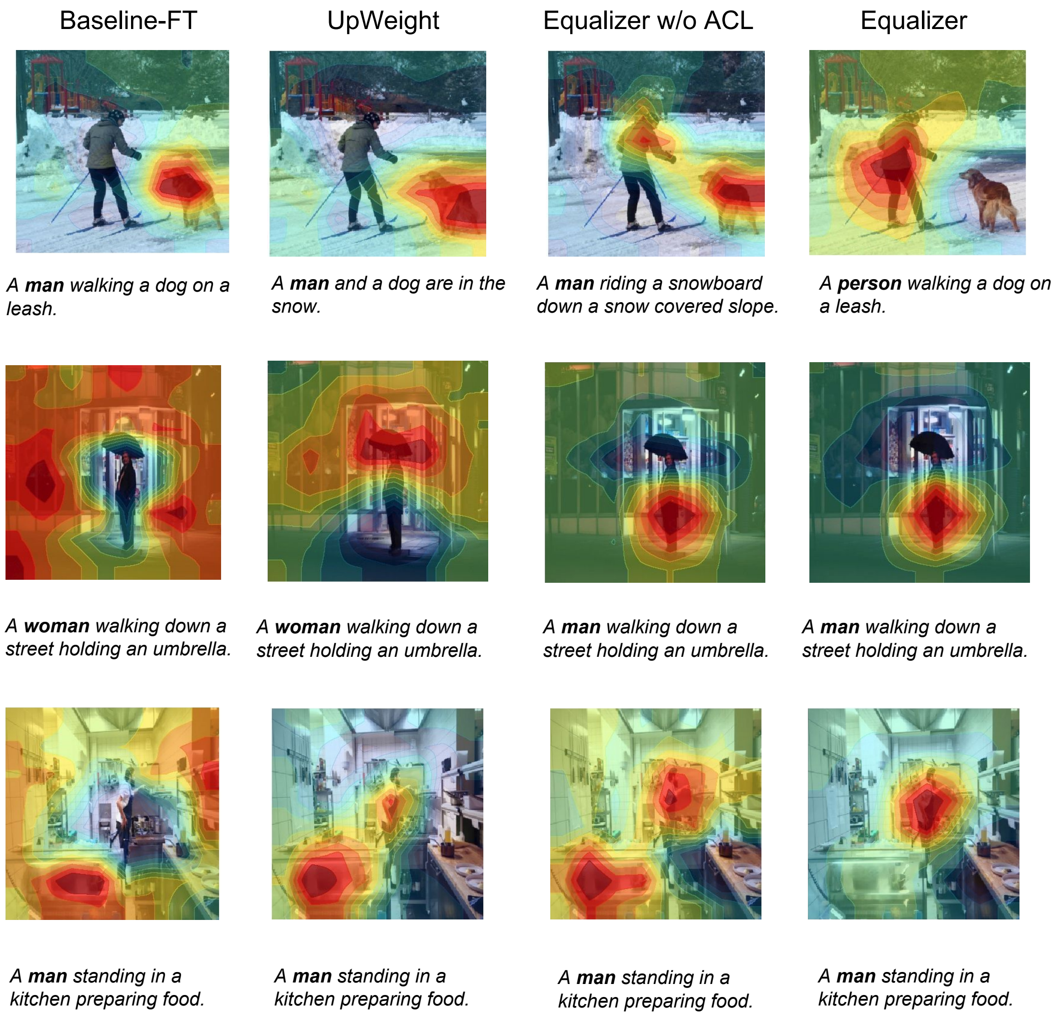
Results on the MSCOCO-Balanced set are presented in Table 4.3 (a) and (b), for the Grad-CAM and saliency maps, respectively. For a fair comparison we provide all models with ground-truth captions. For completeness we also report the random baseline, where the point with the highest weight is selected randomly. We see that Equalizer obtains the best accuracy, significantly improving over the Baseline-FT and all model variants. A similar evaluation on the actual generated captions shows the same trends.

Looking at objects. Using our pointing technique, we can also analyze which MSCOCO objects models are "looking" at when they *do not* point at the person while predicting "man"/"woman". Specifically, we count a "hit" if the highest activation is on an object in question. We compute the following ratio for each gender: number of images where an object is "pointed at" to the true number of images with that object. We find that there are differences across genders, e.g. "umbrella", "bench", "suitcase" are more often pointed at when discussing women, while e.g. "truck", "couch", "pizza" – when discussing men. Our model reduces the overall "delta" between genders for ground truth sentences from an average 0.12 to 0.08, compared to the Baseline-FT. E.g. for "dining table" Equalizer decreases the delta from 0.07 to 0.03.

Qualitative Results. Figure 4.4 compares Grad-CAM visualizations for predicted gender words from our model to the Baseline-FT, UpWeight, and Equalizer w/o ACL. We consistently see that our model looks at the person when describing gendered words. In Figure 4.4 (top), all other models look at the dog rather than the person and predict the gender "man" (ground truth label is "woman"). In this particular example, the gender is somewhat ambiguous, and our model conservatively predicts "person" rather than misclassify the gender. In Figure 4.4 (middle), the Baseline-FT and UpWeight example both incorrectly predict the word "woman" and do not look at the person (women occur more frequently with umbrellas). In contrast, both the Equalizer w/o ACL and the Equalizer look at the person and predict the correct gender. Finally, in Figure 4.4 (bottom), all models predict the correct gender (man), but our model is the only model which looks at the person and is thus "right for the right reasons."

4.5 Discussion

We present the Equalizer model which includes an Appearance ConfusionLoss to encourage predictions to be confused when predicting gender if evidence is obscured and the Confident Loss which encourages predictions to be confident when gender evidence is present. Our Appearance



*A **man** walking a dog on a leash.*

*A **man** and a dog are in the snow.*

*A **man** riding a snowboard down a snow covered slope.*

*A **person** walking a dog on a leash.*

*A **woman** walking down a street holding an umbrella.*

*A **woman** walking down a street holding an umbrella.*

*A **man** walking down a street holding an umbrella.*

*A **man** walking down a street holding an umbrella.*

*A **man** standing in a kitchen preparing food.*

*A **man** standing in a kitchen preparing food.*

*A **man** standing in a kitchen preparing food.*

*A **man** standing in a kitchen preparing food.*

Figure 4.4: Qualitative comparison of multiple baselines and our model. In the top example, being conservative (“person”) is better than being wrong (“man”) as the gender is not obvious. In the bottom example the baselines are looking at the wrong visual evidence.

Confusion Loss, requires human rationales about what is visual evidence is appropriate to consider when predicting gender. We stress the importance of human judgment when designing models which include protected classes. For example, our model can use information about clothing type (e.g., dresses) to predict a gender which may not be appropriate for all applications. Our model requires strong supervision: ground truth annotations for gender evidence must be provided for every image. Interesting future work could consider weaker supervision; either annotating a subset of images with gender evidence, or using outputs from an object detector. Though we concentrate on gender in this work, we believe the generality of our framework could be applied when describing other protected attributes, e.g., race/ethnicity and believe our results suggest Equalizer can be a valuable tool for overcoming bias in captioning models.

Beyond these extensions, an important area of future research is bias discovery. In bias discovery, a system can analyze a dataset or model and automatically determine possible biases. Some initial work has considered automatically discovering bias [MSD18; Dix+18] in areas like visual question answering and text classification. Bias discovery is important because for many models, such as the Equalizer model, the bias must be *known* before it is *mitigated*. Furthermore, understanding biases may allow researchers to adjust their datasets. For example, one well studied bias in VQA is that models can rely on questions (without looking at the image) to provide an answer. Since this bias is known, researchers have placed emphasis on collecting VQA datasets without this bias [Goy+17; Agr+17]. Though we saw in this chapter that training with balanced data does not always reduce bias in machine learning models, awareness of bias can modify how we analyze our data (e.g., in this chapter we consider a variety of metrics beyond sentence accuracy). Finally, some biases can be considered helpful. For example, the kite in Figure 4.2 is easier to recognize as a kite (and not a blanket) because it is in the sky and the person below is in a specific pose. Creating systems where humans can interact with a bias discorvery system and decide which biases are acceptable could mitigate harmful bias without losing the gains of helpful context.

In order to understand if our model was right for the right reason, we relied on visual explanation systems which identified which parts of an image are important for a decision. However, explainable AI is a very active area of research and as we continue to build new models, we also discover serious shortcomings of popular explanation methods such as saliency methods [Ade+18] and attention [JW19]. Thus, progress in explainable AI should allow researchers to more confidently find and mitigate bias in datasets and models. In the next chapter of this thesis we will further explore explainability. In particular, we will consider *textual* explanations which enable systems to justify their decisions with natural language.

Chapter 5

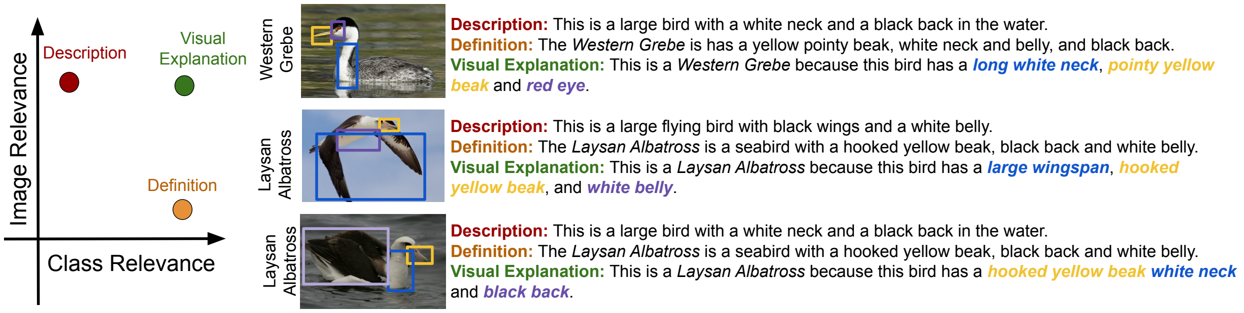
Generating Visual Explanations

5.1 Problem Statement

So far in this thesis we have considered generating text about an image without considering what kinds of systems might benefit from the ability to output text. In this chapter, we consider how we can build on captioning systems in order to build *explanation* systems. Explaining why the output of a visual system is compatible with visual evidence is a key component for understanding and interacting with AI systems [BM14]. Deep classification methods have had tremendous success in visual recognition [KSH12; Gao+16; Don+14], but their outputs can be unsatisfactory if the model cannot provide a consistent justification of why it made a certain prediction. In contrast, systems which can justify why a prediction is consistent with visual elements to a user are more likely to be trusted [TS81]. Explanations of visual systems could also aid in understanding network mistakes and provide feedback to improve classifiers. [1]

We argue that visual explanations must satisfy two criteria: they must be *class discriminative* and *accurately describe* a specific image instance. As shown in Figure 5.1, explanations are distinct from *descriptions*, which provide a sentence based only on visual information, and *definitions*, which provide a sentence based only on class information. Unlike descriptions and definitions, visual explanations detail why a certain category is appropriate for a given image while only mentioning image relevant features. For example, consider a classification system that predicts a certain image belongs to the class “western grebe” (Figure 5.1, top). A standard captioning system might provide a description such as “This is a large bird with a white neck and black back in the water.” However, as this description does not mention *discriminative* features, it could also be applied to a “laysan albatross” (Figure 5.1, bottom). In contrast, we propose to provide *explanations*, such as “This is a western grebe because this bird has a long white neck, pointy yellow beak, and a red eye.” The explanation includes the “red eye” property, which is important for distinguishing between “western grebe” and “laysan albatross”. As such, our system explains *why* the predicted category is the most appropriate for the image.

[1] This chapter is based on joint work done with Marcus Rohrbach, Jeff Donahue, Bernt Schiele, Ronghang Hu, Trevor Darrell, and Zeynep Akata [Hen+16b; Hen+18a] presented at ECCV 2016 and ECCV 2018.



Description: This is a large bird with a white neck and a black back in the water.
Definition: The *Western Grebe* is has a yellow pointy beak, white neck and belly, and black back.
Visual Explanation: This is a *Western Grebe* because this bird has a ***long white neck***, ***pointy yellow beak*** and ***red eye***.

Description: This is a large flying bird with black wings and a white belly.
Definition: The *Laysan Albatross* is a seabird with a hooked yellow beak, black back and white belly.
Visual Explanation: This is a *Laysan Albatross* because this bird has a ***large wingspan***, ***hooked yellow beak***, and ***white belly***.

Description: This is a large bird with a white neck and a black back in the water.
Definition: The *Laysan Albatross* is a seabird with a hooked yellow beak, black back and white belly.
Visual Explanation: This is a *Laysan Albatross* because this bird has a ***hooked yellow beak*** ***white neck*** and ***black back***.

Figure 5.1: Our proposed model generates *explanations* that are both image relevant and class relevant. In contrast, *descriptions* are image relevant, but not necessarily class relevant, and *definitions* are class relevant but not necessarily image relevant.

In addition to discussing discriminative evidence, it is also important that the explanation reflects the actual image content. In order to ensure our explanations are *image relevant*, we *ground* explanatory evidence such as “yellow beak” into the original image. Grounding visual evidence enhances the explanation by adding a visual component to the explanation, but also ensures that the explanation model is not just memorizing discriminative features (e.g., if all western grebe's have red eyes, our model might just learn to discuss the “red eye” attribute without reflecting on the original input).

Our early work [Hen+16b] pioneered text based explanations (e.g., the text generated in Figure 5.1). Our proposed explanation sampler was built off the description model proposed in [Don+15] and was trained with a novel discriminative loss which encourages generated sentences to include class discriminative information; i.e., to be class specific. One challenge is that class specificity is a global sentence property: e.g., while a sentence “This is an all black bird with a bright red eye” is class specific to a “Bronzed Cowbird”, words and phrases in this sentence, such as “black” or “red eye” are less class specific on their own. Our final output is a sampled sentence, so we backpropagate the discriminative loss through the sentence sampling mechanism via a technique from the reinforcement learning literature [Wil92].

More recently [Hen+18a] we extended on our earlier work by proposing a phrase critic to ground explanatory evidence into the original image. Our phrase critic ensures that sentences are image relevant by grounding visual evidence in the image. Individual explanatory phrases in the explanation (e.g., “red eye”) are first grounded in the original input image, then the phrase critic assigns a score to the explanation indicating how relevant it is to the image. Our phrase critic not only encourages sentences to be more image relevant, but produces an additional output in the form of grounded regions that can be helpful for understanding a decision.

We first describe our overall explanation framework, which integrates our work from [Hen+16b] and [Hen+18a]. We then evaluate our model and demonstrate that our novel discriminative loss does indeed lead to more discriminative explanations. Additionally, our phrase critic leads to more image relevant explanations that are grounded in the input image. In addition to leading to more image relevant explanations, grounding visual evidence in an image provides a framework to generate a new kind of explanation, counterfactual explanations, in which a text output indicates how a decision might change if attributes in the image change. We verify our results using both automatic metrics and human evaluations. We finally ask if our explanations can be helpful to humans. In particular, we show that humans are better able to judge whether they should accept or reject an AI decision if provided with an explanation.

5.2 Related Work

In this section, we review recent papers in the context of explanations, mainly focusing on textual and visual explanations. Next, as our proposed explanation sampler relies on REINFORCE [Wil92] to generate sentences, we consider other work which uses reinforcement learning for computer vision problems. Finally, we discuss pragmatics oriented language generation papers that are relevant to out phrase-critic.

Explanations. The importance of explanations for humans has been studied in the field of psychology [Lom12; Lom06], showing that humans use explanations as a guide for learning and understanding by building inferences and seeking propositions or judgments that enrich their prior knowledge. Humans usually seek explanations that fill the requested gap depending on prior knowledge and goal in question. Moreover, explanations are typically contrastive. Much of these ideas are built with careful empirical work, i.e. with human subjects on a specific aspect of explanations [Pac+13]. Since explanations are intended for a human understander, we emphasize the importance of human evaluation in evaluating the relevance of textual explanations to the image as well as looking for the criteria for what makes an explanation good.

Within the artificial intelligence community, automatic reasoning and explanation has a long and rich history[BM14; SB75; Lan+05; Cor+06; VLFM04; Lom+12; LD02; Joh94]. Explanation systems span a variety of applications including explaining medical diagnosis [SB75], simulator actions [Lan+05; Cor+06; VLFM04; Joh94], and robot movements [Lom+12]. Many of these systems are rule-based [SB75] or solely reliant on filling in a predetermined template [VLFM04]. Methods such as [SB75] require expert-level explanations and decision processes. As expert explanations or decision processes are not available during training, our model learns purely from visual features and fine-grained visual descriptions to fulfill our two proposed visual explanation criteria. In contrast to systems like [SB75; Lan+05; Cor+06; VLFM04; Lom+12; LD02] which aim to explain the underlying mechanism behind a decision, Biran et al. [BM14] concentrate on why a prediction is justifiable to a user. Such systems are advantageous because they do not rely on user familiarity with the design of an intelligent system in order to provide useful information. Like [BM14], we aim to generate *rationalizations* explaining the evidence for a decision as opposed to introspective explanations which aim to explain the intermediate activations of neural networks.

Visual Description. Natural language is an intuitive way for humans to interact with artificial agents. Thus, we focus on textual explanations and base our textual explanation systems off of recent advancements in visual description models. Early image description methods rely on detecting visual concepts (e.g., subject, verb, and object) before generating a sentence with either a simple language model or sentence template [Kul+13; Gua+13]. Recent deep models [Vin+15;

Don+15; KFF15; Xu+15a; KSZ14; Fan+15; Mao+15a] outperform such systems and produce fluent, accurate descriptions. Though most description models condition sentence generation only on image features, [Jia+15] condition generation on auxiliary information, such as words used to describe a similar image in the train set. In order to generate explanations, we condition our explanations on category labels.

LSTM sentence generation models are generally trained with a cross-entropy loss between the probability distribution of predicted and ground truth words [Vin+15; Don+15; KFF15; Xu+15a; Mao+15a]. Frequently, however, the cross-entropy loss does not directly optimize for properties desirable at test time. [Mao+16] proposes a training scheme for generating unambiguous region descriptions which maximizes the probability of a region description while minimizing the probability of other region descriptions. Our explanation sampler is trained with a novel loss function for sentence generation which allows us to specify a global constraint on generated sentences.

Textual and Visual Explanation. Our explanation sampler, first proposed in [Hen+16b], was one of the first neural network explanation models which generated text to justify a decision. However, it does not ground the relevant object parts in the sentence or the image. In [HP+18; Kim+18], although an attention based explanation system is proposed, there are no constraints to ensure the actual presence of the mentioned attributes or entities in the image. [WM18] generate explanations by jointly training a visual question answering (VQA) system and text generation system. Spatial attention between the question answering system and text generation is shared, allowing for individual phrases in the explanation (e.g., “cat”) to be tied to spatial regions in the image. Consequentially, albeit generating convincing textual explanations, [Hen+16b; Kim+18; HP+18; WM18] do not include a process for networks to correct themselves if their textual explanation is not well-grounded visually. Additionally, though attention maps can provide insight into which *spatial locations* are important for a decision, when explaining a “long beak” versus a “short beak” we would expect the attention maps to be similar (both focussing on the beak). By relying on a grounding mechanism instead, our system can output a score that indicates how well a phrase “long beak” can be grounded in an image in comparison to the phrase “short beak”. In contrast, we propose a general process to first check whether explanations are accurately aligned with image input and then improve textually explanations by selecting a better-aligned candidate.

Other work has considered *visual explanations* which visualize which regions of an image are important for a decision[FV17; Sel+17; ZF14; Zin+17; PDS18]. Our model produces bounding boxes around regions which correspond to discriminative features, and is thus visual in nature. However, in contrast to visual explanation work, our goal is to rank generated explanatory phrases based on how well they are grounded in an image.

Many vision methods focus on discovering visual features or activated neurons which can help “explain” an image classification decision [BB13; Jia+16; Doe+12; Bau+17; Che+18a]. Importantly, these models do not link discovered discriminative features to natural language expressions. We believe that the methods discovering discriminative visual features are complementary to our proposed system. In fact, discriminative visual features could be used as additional inputs to our model to produce better explanations.

Though we focus on explanations of visual decisions, others have aimed to explain other systems via natural language. For example, [Ehs+18] consider text to help explain decisions made by an AI agent in the video game Frogger, and [Blu+18] consider textual explanations for the task of entailment in natural language phrases.

Fine-grained Classification. Object classification, particularly fine-grained classification, is an attractive setting for explanation systems because describing image content does not suffice as an explanation. Explanation models must focus on aspects that are both class-specific and depicted in the image.

Most fine-grained zero-shot and few-shot image classification systems use attributes [LNH14] as auxiliary information. Attributes discretize a high dimensional feature space into simple and readily interpretable decision statements that can act as an explanation. However, attributes have several disadvantages. They require experts for annotation which is costly and results in attributes which are hard for non-experts to interpret (e.g., "spatulate bill shape"). Attributes are not scalable as the list of attributes needs to be revised to ensure discriminativeness for new classes. Finally, attributes do not provide a natural language explanation like the user expects. We therefore use natural language descriptions [Ree+16b] which achieved superior performance on zero-shot learning compared to attributes and also shown to be useful for text to image generation [Ree+16a].

Reinforcement Learning in Computer Vision. Vision models which incorporate algorithms from reinforcement learning, specifically how to backpropagate through a sampling mechanism, have been applied to visual question answering [And+16b] and activity detection [Yeu+16]. Additionally, [Xu+15a] use a sampling mechanism to attend to specific image regions for caption generation, but use the standard cross-entropy loss during training.

Our explanation sampler was one of the first methods to propose reinforcement learning to optimize for a global sentence property (in our case class discriminativeness) [Hen+16b]. Contemporaneous with [Hen+16b], [Ran+16] proposed training with reinforcement learning to directly optimize standard evaluation metrics such as BLEU [Pap+02]. Others have also considered using reinforcement learning to directly optimize for standard sentence metrics. [Ren+17] explores a better baseline for the reinforcement learning algorithm for sentence generation and [Liu+17] which considers Monte Carlo rollouts for easier optimization. Backpropagating through the sentence sampling mechanism has also been used by visual description systems which aim to optimize and adversarial loss [Dai+17; She+17b].

Pragmatics-Oriented Language Generation. Our work is also related to the recent work of pragmatics-oriented language generation [AK16] where a describer produces a set of sentences, then a choice ranker chooses which sentence best fulfills a specific goal, e.g. distinguishing one image from another. Similarly, image descriptions are generated to make the target image distinguishable from a similar image in [Ved+17], and referential expressions are generated on objects in a discriminative way such that one can correctly localize the mentioned object from the generated expression in [Mao+16]. In this work, we generate textual explanation to maximize both class-specificity and image-relevance. Though similar in spirit, part of our novelty lies in how we learn to rank sentences.

Evaluating Explanations. Quantitative evaluation of explanation systems is by no means straightforward, with different explanation modalities requiring different types of evaluations. In addition to qualitative evaluations, one popular kind of evaluation compares generated explanations to human explanations. For visual explanations researchers have considered metrics such as Earth

Mover's Distance [HP+18], correlation [HP+18] [Das+17a], and the pointing game [Ram+17]. Similarly, for textual explanations researchers have relied on common language generation metrics like CIDEr [VLZP15] and METEOR [BL05].

However, simply considering how well explanations align with a human expert annotation may miss certain aspects of the explanation system. For example, through careful analysis [Ade+18] demonstrate that saliency based visual explanation systems (like Grad-CAM [Sel+17]) fail a variety of "sanity checks" (e.g., random weights and trained models output similar explanations), implying that even if the output aligns well with human explanations, we may not trust the explanation system. [JW19] shows that attention maps are also not necessarily explanatory either.

[DVK18] suggests three types of explanation evaluations: application grounded evaluations (e.g., deploy the explanation system in a real world task and measure if it helps humans perform better), human grounded metrics (e.g., where a human measures the quality of an explanation), and functional evaluations (e.g., automatic metrics which measure how well explanations perform for some proxy evaluation metric). We aim to explore a particular type of explanations (textual explanations), but do not consider a specific end task. Thus we focus on functional evaluations and human grounded metrics. We consider a variety of functional evaluations, such as measuring how well phrases are grounded in an image and how well explanations align with human sentences. For our human grounded metrics, we posit that explanations should be discriminative and image relevant, and directly ask humans to assess our explanations against these criteria. Additionally, we consider a proxy task where humans are asked if they trust the AI decision to make the correct decision given an explanation. Though it might seem obvious that explanations should aid humans in such a task, similar analysis on other explanation systems have not been able to show that explanations consistently help humans with this task [Cha+18].

5.3 Model

Our natural language explanation model incorporates our work from [Hen+16b] and [Hen+18a]. It consists of the following modules illustrated in our system diagram (Figure 5.2):

1. Finegrained classifier (top left). Any off-the-shelf finegrained classifier can be used. Features from the classifier are used in both the explanation sampler and phrase-critic modules.

2. Explanation sampler (middle left). Our explanation sampler [Hen+16b] samples possible textual explanations that can explain why the input belongs to a specific class.

3. Evidence grounding model (bottom left). The evidence grounding model takes possible explanations as input, extracts explanatory phrases (e.g., "blue beak" or "red tail") and outputs a bounding box which indicates where the evidence occurs in the image.

4. Phrase-critic (right). The phrase critic [Hen+18a] takes as input the input image, explanation and grounded evidence and determines if the explanation is well grounded in the original image. This ensures that our final explanations are image relevant, and also provides additional explanatory information in the form of grounded evidence.

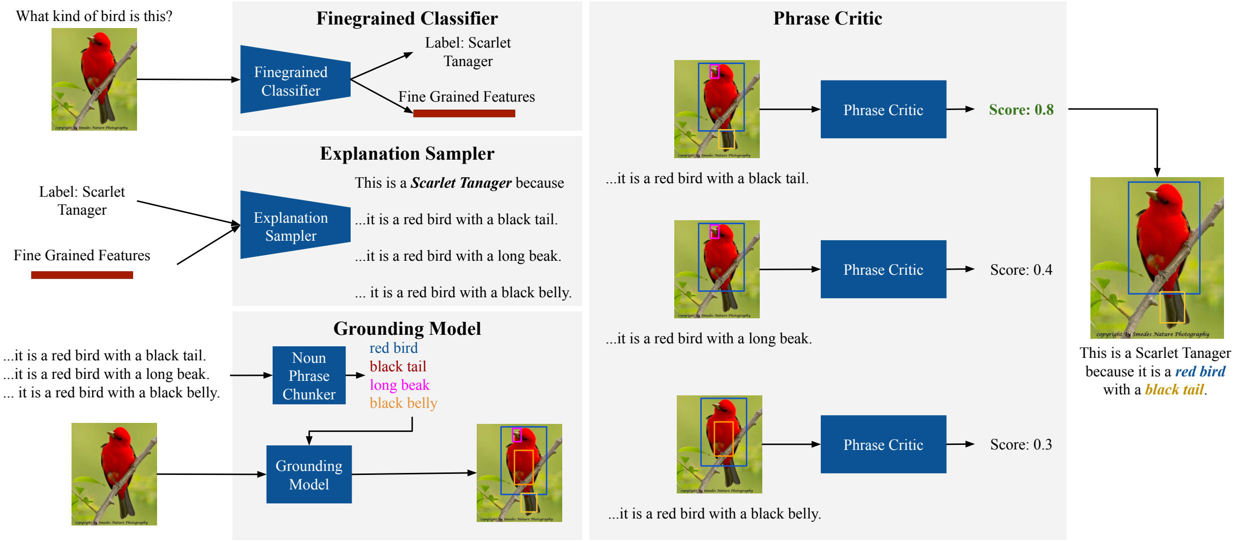

What kind of bird is this?
Finegrained Classifier
Finegrained Classifier
Label: Scarlet Tanager
Fine Grained Features
Explanation Sampler
Label: Scarlet Tanager
Fine Grained Features
Explanation Sampler
This is a Scarlet Tanager because
...it is a red bird with a black tail.
...it is a red bird with a long beak.
... it is a red bird with a black belly.
Grounding Model
...it is a red bird with a black tail.
...it is a red bird with a long beak.
... it is a red bird with a black belly.
Noun Phrase Chunker
red bird
black tail
long beak
black belly
Grounding Model
Phrase Critic
Phrase Critic
Score: 0.8
...it is a red bird with a black tail.
Phrase Critic
Score: 0.4
...it is a red bird with a long beak.
Phrase Critic
Score: 0.3
...it is a red bird with a black belly.
This is a Scarlet Tanager because it is a red bird with a black tail.

Figure 5.2: System diagram. Our explanation model includes the following components: a fine-grained classifier (top left), an explanation sampler (middle left), an evidence grounding model (bottom left), and the phrase critic model (right). See text for details.

The explanation which is best grounded by the phrase-critic is output by the system. The fine-grained classifier and evidence grounding model can be built from any existing classifier or grounding model. In this work, we use the compact bilinear pooling classifier proposed in [Gao+16] and the grounding model proposed in [Hu+17b].The novelty of our work comes from the explanation sampler and phrase-critic. In the following sections, we first detail our explanation sampler and phrase-critic, then discuss the finegrained classifier and grounding model we use in this work.

5.3.1 Explanation Sampler

Our visual explanation model (Figure 5.3) from [Hen+16b] aims to produce an explanation which describes visual content present in a specific image instance which can justify why the image belongs to a specific category. To learn to generate sentences, we introduce a *discriminative loss* (Figure 5.3, top right), which rewards sentences for producing more class relevant text in addition to the standard cross entropy loss (Figure 5.3, bottom right). Our discriminative loss acts on sampled word sequences during training, and enables us to enforce global sentence constraints on sentences. By applying our loss to sampled sentences, we ensure that the final output of our system fulfills our explanation criteria. This is in contrast to the standard cross entropy loss (Figure 5.3, bottom right) usually employed to train caption models which aligns each predicted word to ground truth words, without any notion of which words are the most important to discriminate different categories from each other.

Base Model. Our model is based on LRCN [Don+15], which consists of a convolutional network, which extracts high level visual features, and two stacked recurrent networks (specifically

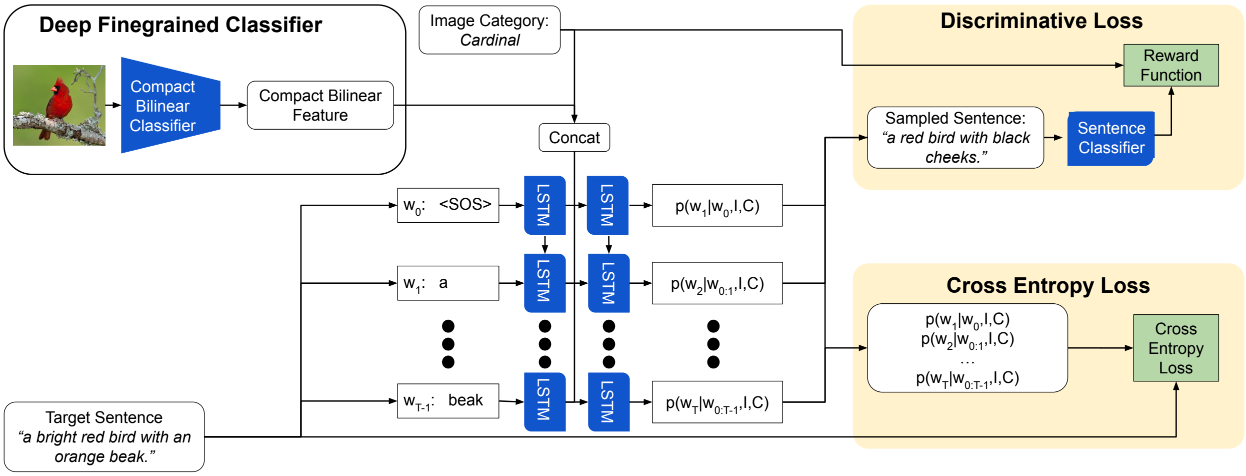


Figure 5.3: Training our explanation model. Our explanation model differs from other caption models because it (1) includes the object category as an additional input and (2) incorporates a reinforcement learning based discriminative loss.

LSTMs), which generate descriptions conditioned on visual features. During inference, the first LSTM receives the previously generated word w_{t-1} as input and produces an output l_t. The second LSTM, receives the output of the first LSTM l_t and an image feature f and produces a probability distribution $p(w_t)$ over the next word. The word w_t is generated by sampling from the distribution $p(w_t)$. Generation continues until an "end-of-sentence" token is generated.

We propose two modifications to the LRCN framework to increase the image relevance of generated sequences (Figure 6.8, top left). First, category predictions are used as an additional input to the second LSTM in the sentence generation model. Intuitively, category information can help inform the caption generation model which words and attributes are more likely to occur in a description. For example, category level information can help the model decide if a red eye or red eyebrow is more likely for a given class. We experimented with a few methods to represent class labels, and found that training a language model, e.g., an LSTM, to generate word sequences conditioned on images, then using the average hidden state of the LSTM across all sequences for all classes in the train set as a vectorial representation of a class works best. Second, we use rich category specific features [Gao+16] to generate relevant explanations.

Each training instance consists of an image, category label, and a ground truth sentence. During training, the model receives the ground truth word w_t for each time step $t \in T$. We define the relevance loss for a specific image (I) and caption (C) as:

$$L_R(I, C) = \frac{1}{N} \sum_{n=0}^{N-1} \sum_{t=0}^{T-1} \log p(w_{t+1}|w_{0:t}, I, C) \tag{5.1}$$

where w_t is a ground truth word and N is the batch size. By training the model to predict each word in a ground truth sentence, the model learns to generate grammatically fluent sentences which reflect the image content. However, this loss does not explicitly encourage generated sentences to discuss discerning visual properties. In order to generate sentences which are category specific, we include a discriminative loss to focus sentence generation on discriminative visual properties of the object.

Discriminative Loss. Our discriminative loss is based on a reinforcement learning paradigm for learning with layers which require sampling intermediate activations of a network. In our formulation, we first sample a sentence and then use the sampled sentence to compute a discriminative loss. By sampling the sentence before computing the loss, we ensure that sentences sampled from our model are more likely to be class specific. Our reinforcement based loss enables us to backpropagate through the sentence sampling mechanism.

We minimize the following overall loss function with respect to the explanation network weights W:

$$L_R(I, C) - \lambda \mathbb{E}_{\tilde{w} \sim p(w|I,C)} \left[R_D(\tilde{w}) \right] \tag{5.2}$$

which is a linear combination of the relevance loss L_R and the expectation of the negative discriminator reward $-R_D(\tilde{w})$ over descriptions $\tilde{w} \sim p(w|I, C)$, where $p(w|I, C)$ is the model's estimated conditional distribution over descriptions w given the image I and category C. Since $\mathbb{E}_{\tilde{w} \sim p(w|I,C)} \left[R_D(\tilde{w}) \right]$ is intractable, we estimate it at training time using Monte Carlo sampling of descriptions from the categorical distribution given by the model's softmax output at each timestep. The sampling operation for the categorical distribution is non-smooth in the distribution's parameters $\{p_i\}$ as it is a discrete distribution. Therefore, $\nabla_W R_D(\tilde{w})$ for a given sample $\tilde{w}$ with respect to the weights W is undefined.

Following the REINFORCE [Wil92] algorithm, we make use of the following equivalence property of the expected reward gradient:

$$\nabla_W \mathbb{E}_{\tilde{w} \sim p(w|I,C)} \left[R_D(\tilde{w}) \right] = \mathbb{E}_{\tilde{w} \sim p(w|I,C)} \left[R_D(\tilde{w}) \nabla_W \log p(\tilde{w}) \right] \tag{5.3}$$

In this reformulation, the gradient $\nabla_W \log p(\tilde{w})$ is well-defined: $\log p(\tilde{w})$ is the log-likelihood of the sampled description $\tilde{w}$, just as L_R is the log-likelihood of the ground truth description. However, the sampled gradient term is weighted by the reward $R_D(\tilde{w})$, pushing the weights to increase the likelihood assigned to the most highly rewarded (and hence most discriminative) descriptions. Therefore, the final gradient we compute to update the weights W, given a description $\tilde{w}$ sampled from the model's softmax distribution, is:

$$\nabla_W L_R - \lambda R_D(\tilde{w}) \nabla_W \log p(\tilde{w}). \tag{5.4}$$

$R_D(\tilde{w})$ should be high when sampled sentences are discriminative. We define our reward simply as $R_D(\tilde{w}) = p(C|\tilde{w})$, or the probability of the ground truth category C given only the generated sentence $\tilde{w}$. By placing the discriminative loss after the sampled sentence, the sentence acts as an information bottleneck. For the model to produce an output with a large reward, the generated sentence must include enough information to classify the original image properly.

For the sentence classifier, we train a single layer LSTM-based classification network to classify ground truth sentences. Our sentence classifier correctly predicts the class of unseen validation

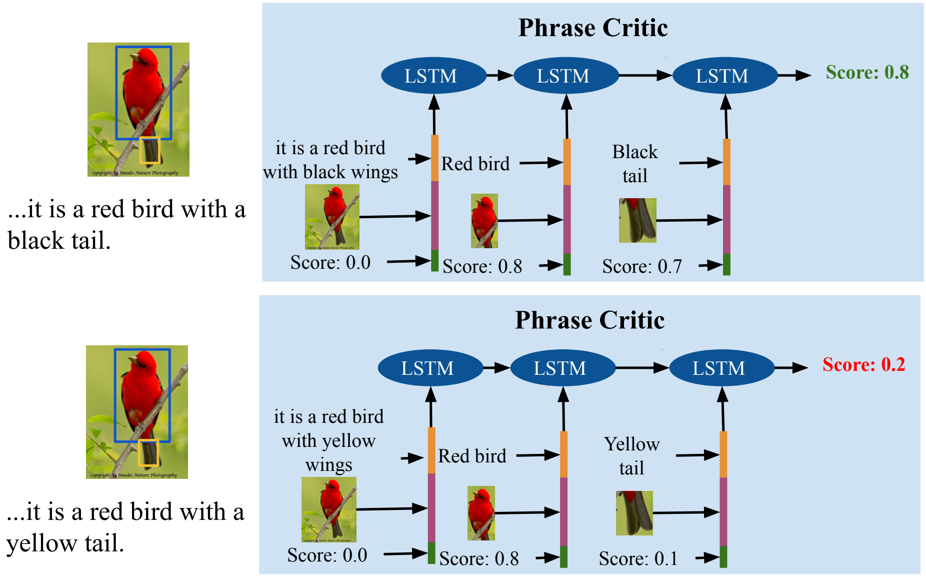


Figure 5.4: Phrase-Critic. Our phrase critic takes as input an attribute (e.g., "black tail"), grounded region, and score from the grounding model and outputs a score which indicates how well the sentence is grounded in the image.

set sentences 22% of the time. This number is possibly low because descriptions in the dataset do not necessarily contain discriminative properties (e.g., "This is a white bird with grey wings." is a valid description but can apply to multiple bird species). Nonetheless, we find that this classifier provides enough information to train our explanation model. Outside text sources (e.g., field guides) could be useful when training a sentence classifier. However, incorporating outside text can be challenging as this requires aligning our image annotation vocabulary to field-guide vocabulary. When training the explanation model, we do not update weights in the sentences classifier.

5.3.2 Phrase Critic

Our explanation sampler is trained to generate discriminative sentences which apply to an image. Often class and image relevance are in opposition. For example, if one attribute frequently occurs within a class, an agent may learn to justify its prediction by mentioning this attribute without even looking at the image. Our phrase critic [Hen+18a] reflects back on the original image, and selects a sampled explanation which is best grounded in the image.

Given a set $\{(\mathcal{A}_i, \mathcal{R}_i, s_i)\}$, where $\mathcal{A}_i$ is an attribute phrase, $\mathcal{R}_i$ is the corresponding region (more precisely, visual features extracted from the region), and s_i the region score, our phrase-critic model, $f_{critic}(\{(\mathcal{A}_i, \mathcal{R}_i, s_i)\})$, maps them into a single image relevance score S_r. For a given attribute phrase A_i such as "black beak", we ground (localize) it into a corresponding image region R_i and obtain its localization score s_i, using an off-the-shelf localization model from [Hu+17b]. It is worth noting that the scores directly produced by the grounding model can not be directly combined with other metrics, such as sentence fluency, because these scores are difficult to normalize across different images and different visual parts. For example, a correctly grounded phrase "yellow belly" may have a much smaller score than the correctly grounded phrase "yellow eye" because a bird belly is less well defined than a bird eye. Henceforth, our phrase-critic model plays an essential role in producing normalized, utilizable and comparable scores. More specifically, given an image I, the phrase-critic model processes the list of $\{(\mathcal{A}_i, \mathcal{R}_i, s_i)\}$ by first encoding each $(\mathcal{A}_i, \mathcal{R}_i, s_i)$ into a fixed-dimensional vector x_{enc} with an LSTM and then applying a two-layer neural network to regress the final score S_r which reflects the overall image relevance of an explanation. As is shown in Figure 5.4, a phrase, grounded region, and score are concatenated and input into the LSTM classifier. Phrases are represented as one hot vectors, and grounded regions are represented by the fc_7 features from the grounding model. Grounded explanations should have a higher score than explanations which cannot be grounded.

We construct ten negative explanation sentences for each image as we explain in the next section. Each negative explanation sentence (not image relevant) gets paired with a positive explanation (image-relevant). We then train our explanation critic using the following margin-based ranking loss $Loss_{rank}$ on each pair of positive and negative explanations, to encourage the model to give higher scores to positive explanations than negative explanations:

$$Loss_{rank} = \max(0, \underbrace{f_{critic}(\{A_i^n\}, I; \theta)}_{S_r^n} - \underbrace{f_{critic}(\{A_i^p\}, I; \theta)}_{S_r^p} + 1) \tag{5.5}$$

where A_i^p are matching noun phrase whereas A_i^n are mismatching noun phrases respectively, therefore S_r^p and S_r^m are the scores of the positive and the negative explanations. In the following, we discuss how we construct our negative image-sentence pairs.

Mining and Augmenting Negative Sentences. The simplest way to sample a negative pair is to consider a mismatching ground truth image and sentence pair. However, we find that mismatching sentences are frequently either too different from ground truth sentences (and thus do not provide a useful training signal) or too similar to ground truth sentences, such that both the positive and negative sentence are image relevant. Hence, inspired by a relative attribute paradigm for recognition and retrieval [PG11], we create negative sentences by flipping attributes corresponding to color, size and objects in attribute phrases. For example, if a ground truth sentence mentions a "yellow belly" and "red head" we might change the attribute phrase "yellow belly" to "yellow beak" and "red head" to "black head". This means the negative sentence still mentions some attributes present in the image, but is not completely correct. We find that creating hard negatives is important when training our self-verification model.

Ranking Explanations. After generating a set of candidate explanations and computing an explanation score, we choose the best explanation based on the score for each explanation. In practice, we find it is important to rank sentences based on both the relevance score S_r and a fluency score S_f (defined as the $\log P(w_{0:T})$). However, we find that first discarding sentences which have a low fluency score, and then choosing the sentence with the highest relevance works better:

$$S = \mathbb{1}\Bigg(\underbrace{\sum_i \log P(w_i|w_{0,\ldots,i-1})}_{S_f} > T\Bigg)\underbrace{f_{critic}(\{A_i\}, I; \theta)}_{S_r} \tag{5.6}$$

where S_r is the relevance score and S_f is the log probability of a sentence based on the trained explanation model. $\mathbb{1}(\cdot)$ is the indicator function and T is a fluency threshold. Including S_f is important because otherwise the explanation scorer will rank "This is a bird with a long neck, long neck, and red beak" high (if a long neck and red beak are present) even though mentioning "long neck" twice is clearly ungrammatical. Based on experiments on our validation set, we set T equal to negative five.

5.3.3 Finegrained Classifier

Our explanation system is agnostic to which finegrained classifier is used for the original classification decision. We use the compact bilinear model [Gao+16] because it is both conceptually straightforward and accurate. The compact bilinear model is built off of the VGG [SZ15] classification model. However, the final two fully connected layers of the model are replaced by a bilinear feature [Car+12], which corresponds to the outer product of the $conv_5$ feature map with itself. As bilinear features are memory intensive, [Gao+16] proposes an estimate to the bilinear feature instead. By replacing the last two layers with the more expressive bilinear feature, we achieve a competitive number on the finegrained bird dataset of 83.9%.

5.3.4 Grounding Model

Our framework for grounding visual features involves two steps: factorizing the sentences into noun phrases and localizing each chunk with a grounding model. In order to verify that explanations are image relevant, for each explanation we extract a list of i attribute phrases ($\mathcal{A}_i$) using a rule-based attribute phrase chunker. Our chunker works as follows: we first use a POS tagger, then extract attribute phrases by finding phrases which syntactically match the structure of attribute phrases. We find that attribute phrases have two basic types of syntactic structure: a noun followed by a verb and an adjective, e.g. "bird is black" or "feathers are speckled", or an adjective (or list of adjectives) followed by a noun, e.g. "red and orange head" or "colorful body". Though this syntactic structure is specific to the bird data, similar methods could be used to extract visual phrases for other applications. Attribute phrases are ordered based on the order in which they occur in the generated visual explanation.

Once we have extracted attribute phrases $\mathcal{A}_i$, we ground each of them to a visual region $\mathcal{R}_i$ in the original image by using [Hu+17b] pre-trained on the Visual Genome dataset [Kri+17] without any access to task-specific ground truth. For a given attribute phrase $\mathcal{A}_i$, the grounding model localizes the phrase into an image region, returning a bounding box $\mathcal{R}_i$ and a score s_i of how likely the returned bounding box matches the phrase. The grounding model works in a retrieval manner. It first extracts a set of candidate bounding boxes from the image, and embeds the attribute phrase into a vector. Then the embedded phrase vector is compared with the visual features of each candidate bounding box to get a matching score (s_i). Finally the bounding box with the highest matching score is returned as the grounded image region. The attribute phrase, the corresponding region, and the region score form an attribute phrase grounding $(\mathcal{A}_i, \mathcal{R}_i, s_i)$. This attribute phrase grounding is used as an input to our phrase-critic.

Whereas visual descriptions are encouraged to discuss attributes which are relevant to a specific class, the grounding model is only trained to determine whether a natural language phrase is in an image. Being discriminative rather than generative, the critic model does not have to learn to generate fluent, grammatically correct sentences, and can thus focus on checking whether the mentioned attribute phrases are image-relevant. Consequently, the models are complementary, allowing one model to catch the mistakes of the other.

5.4 Experiments

5.4.1 Datasets

CUB. We validate our approach on the CUB dataset [Wah+11] which contains 200 classes of fine-grained bird species with approximately 60 images each and a total of 11,788 images of birds. Recently, [Ree+16b] collected sentences for each image with a detailed description of the bird. We note that the descriptions from [Ree+16b] are not provided by bird experts and though [Ree+16b] demonstrated that the text is highly class relevant for the task of zero shot recognition, provided descriptions are not necessarily explanations. However, we find this dataset is suitable for our task as every sentence as well as every image is associated with a single label. Note that CUB does not contain ground truth part bounding boxes, however it contains keypoints that roughly fall on each body part.

FOIL. Our phrase-critic model is flexible and can also be applied to other relevant tasks. To show the generality of our approach, we also consider the dataset proposed in [She+17a] which consists of sentences and corresponding "FOIL" sentences which have exactly one error. [She+17a] proposes three tasks: (1) classifying whether a sentence is image relevant or not, (2) determining which word in a sentence is not image relevant and (3) correcting the sentence error. To use our phrase-critic for (1), we employ a standard binary classification loss. For (2), we follow [She+17a] and determine which words are not image relevant by holding out one word at a time from the sentence. When we remove an irrelevant word, the score from the classifier should increase. Thus, we can determine the least relevant word in a sentence by observing which word (upon removal) leads to the largest score from our classifier. Also following [She+17a], for the third task we replace the foiled word with words from a set of target words and choose a target word based on which one maximizes the score of the classifier. To train our phrase critic, we use the positive and negative samples as defined by [She+17a]. As is done across all experiments, we extract phrases with our noun phrase chunker and use this as input to the phrase-critic.

5.4.2 Baselines

Our two novel contributions are our explanation sampler and our phrase critic. We thus design baselines to ablate both these modules.

Explanation Sampler. Our explanation sampler is trained to generate more class discriminative text. To encourage sampled sentences to be more discriminative, we introduce two techniques. In addition to conditioning our sentence generation on the image input, we condition our sentence generation on the class label as well. We then introduce our discriminative loss. Our first baseline is a *description* model which is equivalent to the LRCN description architecture [Don+15]. To understand how much the class label impacts the explanation, we introduce a *definition* model which generates sentences conditioned only on the class label and is trained with the standard cross entropy loss. Note that in the definition model, the explanations of all image instances which belong to the same class will necessarily be the same. We then consider the *explanation-label* model which is conditioned on both the image input and class label. This model is again trained with only the standard cross entropy loss. To understand how our novel discriminative loss impacts our explanations, we first consider the *explantion-discriminative* model which is the same as the description model, except trained with both the softmax cross entropy loss and discriminative loss. Finally, we compare these baselines and ablations to our *explanation* model which is conditioned on both image features and the class label and is trained with our proposed discriminative loss function.

Phrase Critic. In order to use our phrase critic, we sample many sentence from our explanation sampler and choose the explanation which is most relevant to the input as our final explanation. In our experiments we sample 100 sentences from our explanation sampler. To ablate our phrase critic model we consider three ways to select the best explanation from the sentences output by our explanation sampler. First, we consider outputting the explanation which is most *fluent* according to our explanation model. Our *fluency* baseline outputs a sentence S which maximizes $P(S|I,C)$ where I is the image and C is the category. When phrases from sampled explanations are grounded by the grounding model, the grounding model outputs a bounding box as well as a score. Instead of inputting grounded regions into our phrase critic, we can select sentences based off the average score of all phrases in the explanation. We call this our *average grounding* baseline. Finally, we can score sentences output by our explanation model using the phrase critic.

5.4.3 Metrics

In order to understand how well our explanation model performs, we explicitly measure sentence quality, class discriminativeness, image relevance, and how useful our explanations are to humans when deciding whether to accept or reject an AI decision.

Sentence Quality Metrics. One way to measure explanation quality is to observe how closely our explanations align to the human descriptions used to train our model. We rely on METEOR [BL05] and CIDEr [VLZP15] to measure how well our generated explanations match ground truth sentences in our dataset. Because our ground truth descriptions are not actual explanations, we do not solely rely on these scores to measure the quality of our explanations. Additionally, METEOR and CIDEr are overall sentence quality metrics that do not directly measure if sentence content is discriminative. Thus, we consider these metrics to measure overall sentence quality, but focus on other metrics that explicitly measure class discriminativeness, image relevance, and usefulness to determine whether or not our sentences are good explanations.

Class Relevance. Class relevance measures how well our explanations reflect a specific class label. Our explanation sampler is trained to optimize an LSTM based sentence classifier. We can use the LSTM sentence classifier to measure how discriminative our text explanations are, but this is not a completely fair metric because some models are trained to directly increase the accuracy as measured by the LSTM classifier. Alternatively, we can train our phrase critic as a classifier. We can replace the final layer of the phrase critic which outputs an image relevancy score with a layer that predicts a class label and retrain the phrase critic with a softmax cross entropy classification loss. Note that our explanation sampler models are never explicitly trained to optimize this phrase critic classifier.

In addition to the predicted labels from trained classifiers, we also measure class relevance by considering how similar generated sentences for a class are to ground truth sentences for that class. Sentences which describe a certain bird class, e.g., “cardinal”, should contain similar words and phrases to ground truth “cardinal” sentences, but not ground truth “black bird” sentences. We compute CIDEr scores for images from each bird class, but instead of using ground truth image descriptions as reference sentences, we pool all reference sentences which correspond to a particular class. We call this metric the *class similarity* metric.

Though class relevant sentences should have high class similarity scores, a model could achieve a better class similarity score by producing better overall sentences (e.g., better grammar) without producing more class relevant descriptions. To further demonstrate that our sentences are class relevant, we compute a *class rank* metric. Intuitively, class similarity scores computed for generated sentences about *cardinals* should be higher when compared to *cardinal* reference sentences than when compared to reference sentences from other classes. Consequently, more class relevant models should yield higher rank for ground truth classes. To compute class rank, we compute the class similarity for each generated sentence with respect to each bird category and rank bird categories by class similarity. We report the mean rank of the ground truth class. We emphasize the CIDEr metric because of the TF-IDF weighting over n-grams. If a bird has a unique feature, such as “red eyes”, generated sentences which mention this attribute should be rewarded more than sentences which just mention attributes common across all bird classes. We apply our metrics to images for which we predict the correct label as it is unclear if the best explanatory text should be more similar to the correct class or the predicted class. However, the same trends hold if we apply our metrics to all generated sentences.

Finally, we also conduct a human evaluation in which we directly compare the description model to the explanation model. We posit that if sampled explanations are indeed more discriminative than description systems, human users should be able to more easily understand when an explanation corresponds to a specific class than a description. To test this, we provide humans with either a description or explanation and two images from similar bird classes. We then ask humans to decide which image the sentence is referring to. We can also directly ask humans to determine which sentence best explains a bird classification. However, in order to do this, we require evaluations from people who are familiar with bird classification and thus know which discriminative features are important to discuss. We ask two experienced bird watchers to evaluate 91 images and corresponding sampled sentences from our description, explanation, and baseline and ablation models.

Image Relevance. In order to understand if our phrase critic selects more image relevant sentences we rely on a human evaluation in which we ask humans to judge whether individual noun phrases are present in an image. We additionally measure how well grounded explanatory phrases are in the original image. Though we do not have ground truth bounding boxes for different noun phrases, we do have access to keypoint locations for different bird parts (e.g., “eye” and “beak”). For a given noun phrase (e.g., “red beak”) we can match the noun phrase and grounded region to a keypoint (“beak” in this case) and measure if the keypoint aligns with the grounded region. We measure this in two ways: whether the keypoint falls in the grounded region at all and the Euclidean distance between the center of the grounded region and the keypoint. We also consider the FOIL tasks [She+17a] outlined above to further quantitatively measure the ability of our phrase critic model.

For the sentence quality, class relevance, and image relevance metrics, we only consider generating explanations for the *correct* class because it is unclear what the desired behavior for an incorrect decision is. In the future, we believe this is an important aspect to better understand.

Usefulness. As a final metric, we can ask whether our explanations are helpful to humans. We provide humans with an image, textual explanation, and grounded regions and ask humans whether or not they would accept a classification decision from the AI system given the explanation. As a baseline, we consider an image with no explanation. We provide humans with a training stage where they can observe different images, as well as what the model predicted, if its prediction was correct, and its generated explanation. In this experiment, explanations are produced for the predicted class label. We model this experiment after the experiments presented in [Cha+18] who demonstrated that visual explanations in the terms of saliency maps were *not* helpful to humans when deciding whether or not to accept the decision of an AI system.

5.4.4 Results

In this section, we present our results on explaining fine-grained bird classification. We first evaluate different components of our explanation system, then explore counterfactual explanations, and, finally, conduct a user experiment that demonstrates the usefulness of textual explanations.

5.4.4.1 Explanation Sampler

In this section, we compare the performance of different explanation samplers. We focus on the sentence quality and discriminativeness of our description, definition, explanation-label, explanation-dis, and explanation models.

Sentence Quality. Table 5.1 compares our explanation sampler model and ablations using the METEOR and CIDEr metrics. The definition and description perform similarly when considering sentence quality metrics. However, the explanation-label model performs better than either

	METEOR	CIDEr
Description	27.7	42.0
Definition	27.9	43.8
Explanation-Label	28.1	44.7
Explanation-Dis	28.8	51.9
Explanation	29.2	56.7

Table 5.1: We compare sentence quality of different explanation samplers. We find that by making our explanations more discriminative, our sentences also improve standard sentence quality metrics.

	Similarity	Rank	Acc-D	Acc-PC
Description	35.30	24.43	14.74	63.6
Definition	42.60	15.82	38.08	65.2
Explanation-Label	40.86	17.69	28.70	65.1
Explanation-Dis	43.61	19.80	34.05	64.7
Explanation	52.25	13.12	54.38	65.6

Table 5.2: We compare which training mechanism produces more *class relevant* explanations. Acc-D is the accuracy from the sentence classifier and Acc-PC is accuracy from the phrase critic. Across all proposed metrics, the explanation model performs best.

the description or definition indicating that the class and image information are complementary. Comparing the description model to the explanation-dis model and the explanation-label model to our full explanation model, we can see that the discriminative loss substantially improves sentence quality.

Class Relevance. Table 5.2 compares our different explanation sampler to baselines and ablations on our class relevancy metrics. First, we note that the definition model tends to perform better than a description model indicating it is easier for our model to generate class discriminative sentences when provided with the label information. Unlike the sentence quality metrics, when considering the class relevancy metrics our definition model performs better than the explanation-label model. However, we note that the discriminative loss consistently leads to more class relevant sentences. Importantly, our explanation model performs best on all automatic class relevancy metrics reported in Table 5.2.

Table 5.3 reports our human evaluation in which we ask humans to select which image corresponds to either description or explanation text. We find that it is easier for humans to determine which bird corresponds to explanatory text when text is sampled from our explanation sampler than when text is sampled from our description model indicating that our explanation model produces more discriminative text. Table 5.4 reports which explanations our experienced bird watchers prefer. We ask bird watchers to rank explanatory text from best to worst and then report the average rank of each model. The explanation model has the highest mean rank, indicating our bird experts preferred explanations from our explanation model as opposed to baselines and ablations.

	Correct Image Chosen
Description	52.0
Explanation	56.0

Table 5.3: Humans were provided text from the description or explanation models and were required to use the text to select which of two images corresponded to the image. Humans were better able to match images to text with the explanation model, indicating that the explanation text included more discriminative evidence.

	Mean Rank
Description	3.11
Definition	2.92
Explanation-Label	2.97
Explanation-Dis	3.22
Explanation	2.78

Table 5.4: Experienced bird watchers ranked sentences (from best to worst) from our explanation system as well as baselines and ablations. Lower is better.

Qualitative Results. Figure 5.5 compares sentences generated by our definition and description baselines, explanation-label and explanation-discriminative ablations and explanation model. Each model produces reasonable sentences, however, we expect our explanation model to produce sentences which discuss class relevant attributes. For many images, the explanation model mentions attributes that not all other models mention. For example, in Figure 5.5, row 1, the explanation model specifies that the "bronzed cowbird" has "red eyes" which is a rarer bird attribute than attributes mentioned correctly by the definition and description models ("black", "pointy bill"). Similarly, when explaining the "White Necked Raven" (Figure 5.5 row 3), the explanation model identifies the "white nape", which is a unique attribute of that bird. Based on our image relevance metrics, we also expect our explanations to be more image relevant. An obvious example of this is in Figure 5.5 row 7 where the explanation model includes only attributes present in the image of the "hooded merganser", whereas all other models mention at least one incorrect attribute.

5.4.4.2 Phrase Critic.

In this section we evaluate the performance of our phrase critic. In order to apply our phrase critic, we sample 100 sentences from our explanation sampler mechanism using random sampling [Don+16]. Our evaluations focus on how well generated text is grounded in the original image.

Image Relevancy of Generated Phrases. To measure image relevancy of phrases generated in our textual explanations, we first consider a human evaluation. For explanations selected by our phrase critic as well as the fluency and average grounding baselines, we extract generated noun phrases (e.g., "long beak"). We then ask workers on Amazon Mechanical Turk (AMT) if the noun phrase is present in the image. We consider a phrase to be in an image if two out of three AMT workers agree the phrase is in the image. In Table 5.2 we show results for the percentage of noun phrases which are generated correctly and the percentage of correct sentences which are generated. We consider a sentence to be correct if all noun phrases in the sentence are correct. We see that the phrase critic model outperforms our baselines and ablations. The improvement is most striking

This is a **Bronzed Cowbird** *because ...*
Definition: this bird is **black** with **blue** on its wings and has a long **pointy beak**.
Description: this bird is **nearly all black** with a short **pointy bill**.
Explanation-Label: this bird is **nearly all black** with **bright orange eyes**.
Explanation-Dis.: this is a **black bird** with a **red eye** and a **white beak**.
Explanation: this is a **black bird** with a **red eye** and a **pointy black beak**.

This is a **Black Billed Cuckoo** *because ...*
Definition: this bird has a **yellow belly** and a **grey head**.
Description: this bird has a **yellow belly** and **breast** with a **gray crown** and **green wing**.
Explanation-Label: this bird has a **yellow belly** and a **grey head** with a **grey throat**.
Explanation-Dis.: this is a **yellow bird** with a **grey head** and a **small beak**.
Explanation: this is a **yellow bird** with a **grey head** and a **pointy beak**.

This is a **White Necked Raven** *because ...*
Definition: this bird is **black in color** with a **black beak** and **black eye rings**.
Description: this bird is **black** with a **white spot** and has a **long pointy beak**.
Explanation-Label: this bird is **black** in color with a **black beak** and **black eye rings**.
Explanation-Dis.: this is a **black** bird with a **white nape** and a **black beak**.
Explanation: this is a **black** bird with a **white nape** and a **large black beak**.

This is a **Northern Flicker** *because ...*
Definition: this bird has a **speckled belly and breast** with a **long pointy bill**.
Description: this bird has a **long pointed bill grey throat** and **spotted black and white mottled crown**.
Explanation-Label: this bird has a **speckled belly and breast** with a **long pointy bill**.
Explanation-Dis.: this is a **grey bird** with **black spots** and a **red spotted crown**.
Explanation: this is a **black and white spotted bird** with a **red nape** and a **long pointed black beak**.

This is a **American Goldfinch** *because ...*
Definition: this bird has a **yellow crown** a **short and sharp bill** and a **black wing** with a **white breast**.
Description: this bird has a **black crown** a **yellow bill** and a **yellow belly**.
Explanation-Label: this bird has a **black crown** a **short orange bill** and a **bright yellow breast and belly**.
Explanation-Dis.: this is a **yellow bird** with a **black wing** and a **black crown**.
Explanation: this is a **yellow bird** with a **black and white wing** and an **orange beak**.

This is a **Yellow Breasted Chat** *because ...*
Definition: this bird has a **yellow belly and breast** with a **white eyebrow** and **gray crown**.
Description: this bird has a **yellow breast and throat** with a **white belly and abdomen**.
Explanation-Label: this bird has a **yellow belly and breast** with a **white eyebrow** and **gray crown.**
Explanation-Dis.: this is a bird with a **yellow belly** and a **grey back and head**.
Explanation: this is a bird with a **yellow breast** and a **grey head and back**.

This is a **Hooded Merganser** *because ...*
Definition: this bird has a **black crown** a **white eye** and a **large black bill**.
Description: this bird has a **brown crown** a **white breast** and a **large wingspan**.
Explanation-Label: this bird has a **black and white head** with a large **long yellow bill** and **brown tarsus and feet.**
Explanation-Dis.: this is a **brown bird** with a **white breast** and a **white head**.
Explanation: this bird has a **black and white head** with a **large black beak**.

Figure 5.5: Example sentences generated by our baseline models, ablation models, and proposed explanation model. Correct attributes are highlighted in green, mostly correct attributes are highlighted in yellow, and incorrect attributes are highlighted in red. The explanation model consistently discusses image relevant and class relevant features.

for percentage correct sentences. This is perhaps because the phrase critic is directly optimized to determine whether or not sentences are image relevant.

Phrase Grounding. As the CUB dataset does not contain ground-truth bounding boxes, we cannot evaluate the precision of our detected part bounding boxes w.r.t. a ground truth. However, the dataset contains keypoints for 15 body parts, e.g. bill, throat, left eye, nape, etc. and utilizing these keypoint annotations that roughly correspond to “beak”, “head”, “belly” and “eye” regions,

Method	% Correct Noun Phrases	% Correct Sentences
Fluency	76.64	52.10
Average grounding	76.32	49.85
Our Phrase Critic	**77.96**	**61.97**

Table 5.5: Human evaluations comparing reranking sampled explanations using fluency, average grounding, and our phrase critic model. We consider the percentage of correct noun phrases and the correct sentences.

provides us a good proxy for this task. We measure how frequently a keypoint falls into the detected bounding box of the corresponding body part to determine the accuracy of the bounding boxes. In addition, we measure the distance of the corresponding keypoint to the center of the bounding box to determine the precision of the bounding boxes. Note that for the results in the first row, we take the explanation generated by [Hen+16b] and ground the phrases using the off-the-shelf grounding model [Hu+17b].

Our results in Figure 5.6 show that while "beak", "head" and "belly" regions are detected with high accuracy (95.88%, 74.06% and 66.65% resp.), "eye" detections are weaker (56.72%). When we look at the distance between the bounding box center and the keypoint, we observe a similar trend. The head region gets detected by our model significantly better than others, i.e. 20.26 vs 46.31 with [Hu+17b] and 57.56 with [Hen+16b]. The belly and the beak distances are close to the ones measured by the grounding model whereas the eye region gets detected with a lower precision with our model compared to the grounding model.

We closely investigate the accuracy of the predicted noun phrases that fall into the eye region and observe that although the eye regions get detected with a higher precision with the baseline grounding model, the semantic meaning of the attribute gets predicted more accurately with our phrase critic. For instance, our model mentions "red eye" more accurately than the grounding model although the part box is more accurately localized by the grounding model. One example of this can be seen in Figure 5.6 (top right) where the grounder selects the sentences "... this is a black bird with a white eye and a red eye." Here, the grounding model has selected a sentence which cannot be true (the bird cannot have white eye as well as a red eye). Even though the bounding box around the eye is accurate, the modifying attributes are not both correct.

Qualitative Explanation Results. In Figure 5.6, the results on the left are generated by our phrase critic model, the ones in the middle by the grounding model [Hu+17b] and the ones on the right are by the fluency model. Note that fluency baseline does not contain an attribute phrase grounder, therefore we cannot localize the evidence for the given explanation here. As a general observation, our model improves over both baselines in the following ways. Our critic model (1) grounds attribute phrases both in the image and in the sentence, (2) is in favor of accurate and class-specific noun phrases and (3) provides the cumulative score of each explanatory sentence.

To further emphasize the importance of visual and textual grounding of the noun phrases in evaluating the accuracy of the visual explanation model, let us more closely examine the second row of Figure 5.6. We note that all models mention a "black bird" and "red cheek patch".

	% Accuracy				Euclidean Distance			
Explanations	Beak	Head	Belly	Eye	Beak	Head	Belly	Eye
Fluency	93.50	58.74	65.58	55.11	24.16	57.56	56.80	76.90
Average Grounding	94.30	60.60	65.40	**60.78**	**22.66**	46.31	**52.69**	**57.55**
Phrase Critic	**95.88**	**74.06**	**66.65**	56.72	23.74	**20.26**	**52.75**	69.83

Table 5.6: Evaluating the grounding accuracy for four commonly mentioned bird parts. As we have no have access to ground truth boxes, we measure how frequently the ground truth keypoints fall within a detected bounding box, measuring the % of the keypoints that fall inside the bounding box (left) and the distance between the keypoint and the center of the bounding box (right). The fluency baseline does not include noun phrase grounding in the ranking process, so we apply [Hu+17b] to noun phrases extracted from sentences.

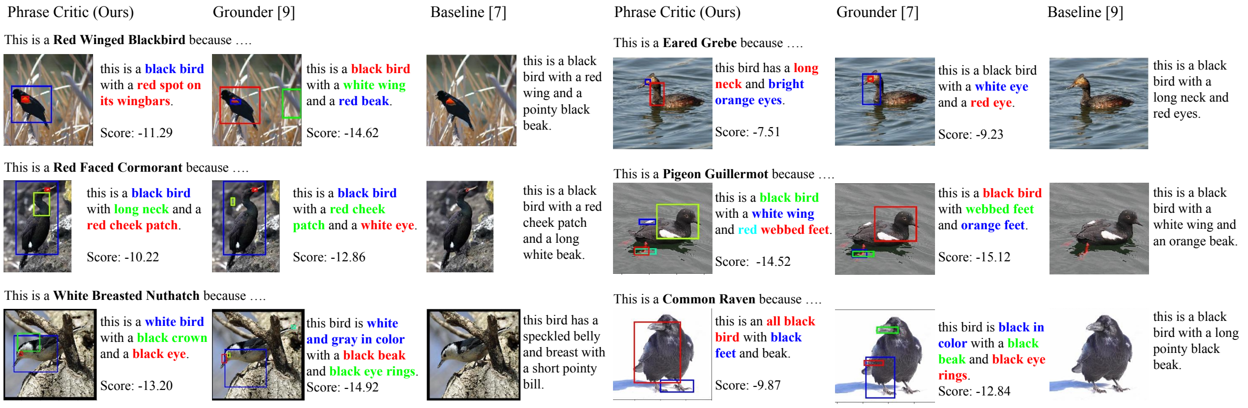

Phrase Critic (Ours)
Grounder [9]
Baseline [7]
This is a **Red Winged Blackbird** because ....
this is a **black bird** with a **red spot on its wingbars**.
Score: -11.29
this is a **black bird** with a **white wing** and a **red beak**.
Score: -14.62
this is a black bird with a red wing and a pointy black beak.
This is a **Red Faced Cormorant** because ....
this is a **black bird** with **long neck** and a **red cheek patch**.
Score: -10.22
this is a **black bird** with a **red cheek patch** and a **white eye**.
Score: -12.86
this is a black bird with a red cheek patch and a long white beak.
This is a **White Breasted Nuthatch** because ....
this is a **white bird** with a **black crown** and a **black eye**.
Score: -13.20
this bird is **white and gray in color** with a **black beak** and **black eye rings**.
Score: -14.92
this bird has a speckled belly and breast with a short pointy bill.
Phrase Critic (Ours)
Grounder [7]
Baseline [9]
This is a **Eared Grebe** because ....
this bird has a **long neck** and **bright orange eyes**.
Score: -7.51
this is a black bird with a **white eye** and a **red eye**.
Score: -9.23
this is a black bird with a long neck and red eyes.
This is a **Pigeon Guillermot** because ....
this is a **black bird** with a **white wing** and **red webbed feet**.
Score: -14.52
this is a **black bird** with **webbed feet** and **orange feet**.
Score: -15.12
this is a black bird with a white wing and an orange beak.
This is a **Common Raven** because ....
this is an **all black bird** with **black feet** and beak.
Score: -9.87
this bird is **black in color** with a **black beak** and **black eye rings**.
Score: -12.84
this is a black bird with a long pointy black beak.

Figure 5.6: Our phrase-critic model generates more image-relevant explanations compared to [Hen+16b] justified by the grounding of the noun phrases. Compared to Grounder [Hu+17b], our phrase-critic generates more class-specific explanations. The numbers indicate the cumulative score of the explanation computed by our phrase-critic ranker.

As the "Red Faced Cormorant" has these properties, these attributes are accurate. However, the explanation sentence is more trustable when the visual evidence of the noun phrase properly localized, which is not done by the baseline explanation model. To verify our intuition that grounded explanations are more trustable, we ask Amazon Mechanical Turke workers to evaluate whether our explanations with or without bounding boxes are more informative. Our results indicate that bounding boxes are informative (41.9% of the time bounding boxes lead to more informative explanations and 49.3% of the time explanations with and without bounding boxes are equally informative). Therefore, we emphasize that visual and textual grounding is beneficial and important for evaluating the accuracy of the visual explanation model.

Again examining the "Red Faced Cormorant" in the second row of Figure 5.6, although "red cheek patch" is correctly grounded both by our phrase critic and the baseline phrase grounder, our

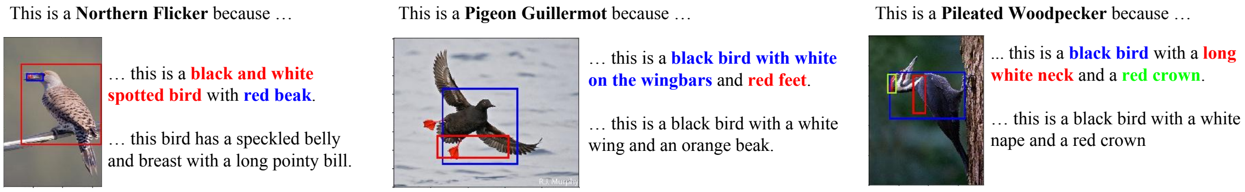

This is a Northern Flicker because …
… this is a black and white spotted bird with red beak.
… this bird has a speckled belly and breast with a long pointy bill.
This is a Pigeon Guillermot because …
… this is a black bird with white on the wingbars and red feet.
… this is a black bird with a white wing and an orange beak.
This is a Pileated Woodpecker because …
... this is a black bird with a long white neck and a red crown.
… this is a black bird with a white nape and a red crown

Figure 5.7: Failure Cases: In some cases our model predicts an incorrect noun phrase and the grounding may reveal the reason. On the other hand, in some cases although the explanation sentences are accurate, the phrases are not grounded well, i.e. the bounding boxes are off. Top: Our phrase-critic, Bottom: Baseline [Hen+16b].

phrase critic also mentions and grounds an important class-specific attribute of "long neck" while the grounding model mentions a missing "white eye" attribute which it cannot ground. Thus, the score based ranking of noun phrase and region pairs lead to more accurate and visually grounded visual explanations.

Thanks to the integrated visual grounding capability and phrase ranking mechanism, the critic is able to detect the mistakes of the baseline model and correct them. Some detailed observations from Figure 5.6 are as follows. "Red Winged Blackbird" having a "red spot on its wingbars" is one of the most discriminative properties of this bird which is mentioned by our critic and also grounded accurately. Similarly, the most important property of "Eared Grebe" is its "red eyes". We see that for "Pigeon Guillermot" our model talks about its "white wing" and "red webbed feet" whereas the grounding model does not mention the "white wing" property and the baseline model does not only ground the phrase but also it does not mention the "red feet". Our model does not only qualitatively generate more accurate explanations, these sentences also get higher cumulative phrase scores as shown beside each image in the figure providing another level of confidence.

In Figure 5.7 we present some typical failure cases of our model. In some cases such as the first example, the nouns, i.e. bird and beak, are correctly grounded however the attribute is wrong. Although the bird has a black beak, due to the red color of the fruit it is holding, our model thinks it is a red beak. Another failure case is when the noun phrases are semantically accurate however they are not correctly grounded. For instance, in the second example, both "black bird with white on the wingbars" and "red feet" are correctly identified, however the bounding box of the feet is off. Note that in CUB dataset, the ground truth part bounding box annotations are not available, hence our model figures out the location of a "red feet" by adapting the grounding model trained on Visual Genome, which may not include similar box-phrase combinations. Similarly, in the third example, the orientation of the bounding box of the phrase "long white neck" is inaccurate since the bird is perching on the tree trunk vertically although most of the birds perch on tree branches in a horizontal manner.

FOIL Experiments. In addition to showing our phrase critic works to generated better visual explanations, we also apply it to the FOIL task. Here we see that our phrase critic can more accurately determine whether a sentence describing an image is image relevant or not.

Table 5.7 shows the performance of our phrase critic on the FOIL tasks compared to the best

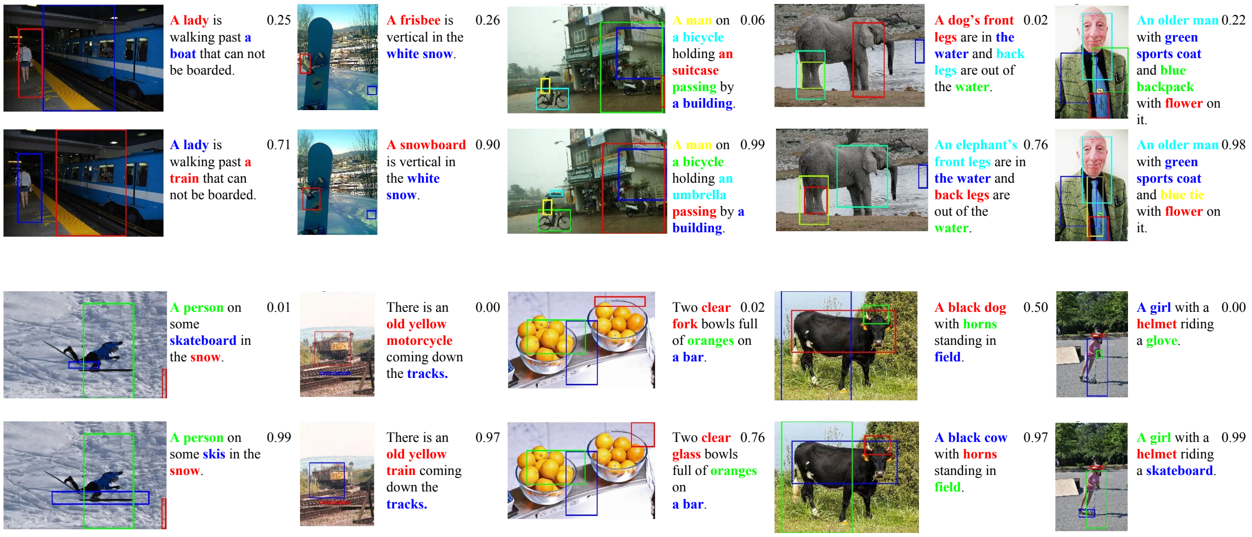

A lady is walking past a boat that can not be boarded. 0.25
A frisbee is vertical in the white snow. 0.26
A man on a bicycle holding an suitcase passing by a building. 0.06
A dog's front legs are in the water and back legs are out of the water. 0.02
An older man with green sports coat and blue backpack with flower on it. 0.22
A lady is walking past a train that can not be boarded. 0.71
A snowboard is vertical in the white snow. 0.90
A man on a bicycle holding an umbrella passing by a building. 0.99
An elephant's front legs are in the water and back legs are out of the water. 0.76
An older man with green sports coat and blue tie with flower on it. 0.98
A person on some skateboard in the snow. 0.01
There is an old yellow motorcycle coming down the tracks. 0.00
Two clear fork bowls full of oranges on a bar. 0.02
A black dog with horns standing in field. 0.50
A girl with a helmet riding a glove. 0.00
A person on some skis in the snow. 0.99
There is an old yellow train coming down the tracks. 0.97
Two clear glass bowls full of oranges on a bar. 0.76
A black cow with horns standing in field. 0.97
A girl with a helmet riding a skateboard. 0.99

Figure 5.8: qualitative foil results: we present the image with foil sentence (top) and correct sentence (bottom) as determined by our phrase-critic model. the numbers indicate our phrase-critic score of the given sentence. by design our model grounds all the phrases in the sentence, including the foil phrases.

	Classification	Detection	Correction
IC - Wang [Wu+17]	42.21	27.59	22.16
HieCoAtt [Lu+16]	64.14	38.79	4.21
Grounding model [Hu+17b]	56.68	39.80	8.80
Phrase Critic (Ours)	**87.00**	**73.72**	**49.60**

Table 5.7: Quantitative FOIL results: Our phrase critic significantly outperforms the state-of-the-art [Wu+17; Lu+16] reported in [She+17a] and the Grounding model [Hu+17b] on all three FOIL tasks.

performing models evaluated in [She+17a]. IC-Wang [Wu+17] is an image captioning model whereas HieCoAtt [Lu+16] is an attention based VQA model. As described above, we follow the protocol of [She+17a] when evaluating our model on the FOIL tasks. To apply the grounding model to the classification task, we determine a threshold score on the train set (i.e., any sentence with an average grounding score above a certain threshold is classified as image relevant).

Our results show that the phrase critic is able to effectively adapt a grounding model in order to determine whether or not sentences are image relevant. We see that our grounding model baseline performs competitively when compared to prior work, indicating that grounding noun phrases is a promising step to determine if sentences are image relevant. However, our phrase critic model outperforms all baselines by a wide margin, outperforming the next best model by over 20 points on the classification task, over 30 points on the word identification task, and close to 30 points on the word correction task. The large gap between the grounding model baseline and the phrase-critic highlights the importance of our phrase-critic in learning how to properly adapt outputs from a grounding model to our final task.

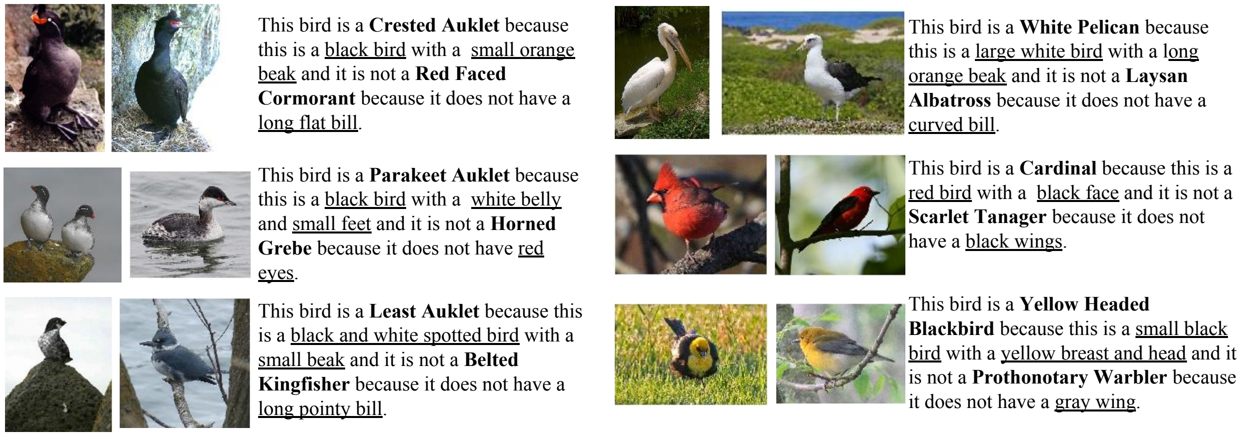
5.8 shows example negative and positive sentences from the FOIL dataset, the grounding determined by our phrase-critic, and the score output by our phrase-critic model. Our general observation is that our phrase critic gives a significantly lower score to FOIL sentences which are not image relevant. In addition, it accurately grounds mentioned objects and accurately scores sentences based on if they are image relevant.

Some detailed observations are as follows. For the first example the score of the FOIL sentence is 0.25 as the sentence contains "a boat" phrase that is inaccurate whereas the sentence with the correct phrase, i.e. "a train", gets the score 0.71 which clearly indicates that this is the correct sentence. Our model is able to ground more than two phrases accurately as well. For the last image in the first row, the phrases "an older man", "green sports coat" and "flower" are correctly predicted and grounded whereas "blue backpack" gets grounded close to the shoulder, which is a sensible region to consider even though there is no backpack in the image. This FOIL sentence gets the score 0.22 whereas the correct sentence that gets the score 0.98 grounds "blue tie" correctly while also correctly grounding all other phrases in the sentence.

When the FOIL object is one of the many objects in the sentence and occupies a small region in the image, our phrase-critic is also successful. For instance in the third image in the first row, "an suitcase" is grounded in an arbitrary location on the side of an image which leads to an extremely low sentence score, 0.06. In the image relevant sentence, "an umbrella" is grounded correctly leading to a high sentence score, 0.99. In conclusion, our phrase critic accurately grounds the phrases when they are present and assigns scores to the matching phrases and bounding boxes that helps us further understand why a model has taken such a decision.

5.4.4.3 Counterfactuals

Another way of explaining a visual concept is through generating *counterfactual* explanations that indicate why the classifier does not predict another class label. To construct counterfactual explanations, we posit that if an attribute is discriminative for another class, i.e. a class that is different from the class that the query image belongs to, but not present in the query image, then this attribute is a *counterfactual* evidence. To discuss counterfactual evidence for a classification decision, we first hypothesize which visual evidence might indicate that the bird belongs to another class. We do so by considering explanations produced by our phrase-critic for visually most similar examples from a different, i.e. counterfactual, classes. Our phrase-critic determines which attributes are most class specific for the counterfactual class and most image relevant for the query image while generating factual explanations. While generating counterfactual explanations, our model determines the counterfactual evidence by searching for the attributes of the counterfactual class which lead to the lowest phrase-critic score for the query image. We then construct a sentence by negating counterfactual phrases. For instance, "bird has a long flat bill" is negated to "bird does not have a long flat bill" where the counterfactual phrase is the "long flat bill". Alternatively, we



This bird is a **Crested Auklet** because this is a black bird with a small orange beak and it is not a **Red Faced Cormorant** because it does not have a long flat bill.

This bird is a **Parakeet Auklet** because this is a black bird with a white belly and small feet and it is not a **Horned Grebe** because it does not have red eyes.

This bird is a **Least Auklet** because this is a black and white spotted bird with a small beak and it is not a **Belted Kingfisher** because it does not have a long pointy bill.

This bird is a **White Pelican** because this is a large white bird with a long orange beak and it is not a **Laysan Albatross** because it does not have a curved bill.

This bird is a **Cardinal** because this is a red bird with a black face and it is not a **Scarlet Tanager** because it does not have a black wings.

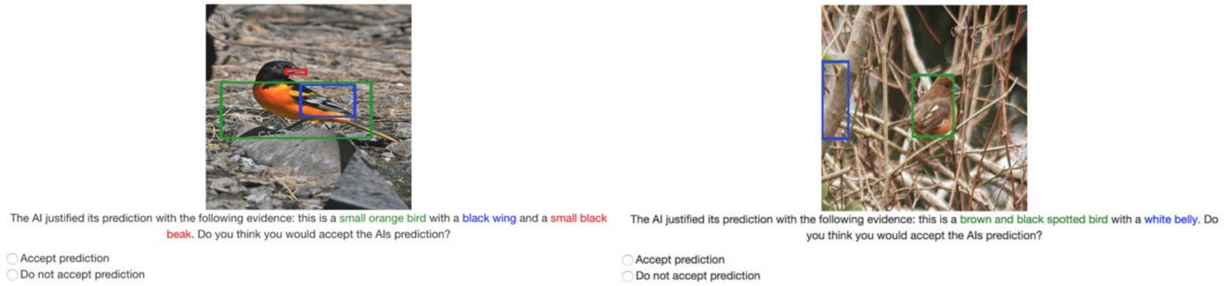
This bird is a **Yellow Headed Blackbird** because this is a small black bird with a yellow breast and head and it is not a **Prothonotary Warbler** because it does not have a gray wing.

Figure 5.9: Our phrase-critic is able to generate factual and counterfactual explanations. Factual explanations mention the characteristic properties of the correct class (left image) and counterfactual explanations mention the properties that are not visible in the image, i.e. non-groundable properties, for the negative class (right image).

can use the same evidence to rephrase the sentence “If this bird had been a (counterfactual class), it would have had a long flat bill.”

To illustrate, we present our results in Figure 5.9. Note that the figures show two images for each result where the first image is the query image. The second image is the counterfactual image, i.e. the most similar image to the query image from the counterfactual class, that we show only for reference purposes. The counterfactual explanation is generated for this image just for determining the most class-specific noun phrase. Once a list of counterfactual noun phrases is determined, those noun phrases are grounded in the query image and the noun phrase that gets the lowest score is determined as the counterfactual evidence. To illustrate, let us consider an image of a *Crested Auklet* and a nearest neighbor image from another class, e.g., *Red Faced Cormorant*. The attributes “black bird” and “long flat bill” are possible counterfactual attributes for the original crested auklet image. We use our phrase-critic to select the attribute which produces the *lowest* score for the Crested Auklet image.

Figure 5.9 shows our final counterfactual explanation for why the *Crested Auklet* image is not a *Red Faced Cormorant* (it does not have a long flat bill). On the other hand, when the query image is a *Parakeet Auklet*, the factual explanation talks about “red eyes” which are present for *Horned Grebe* but not for *Parakeet Auklet*. Similarly, a *Least Auklet* is correctly determined to be a “black and white spotted bird” with a “small beak” while a *Belted Kingfisher* is a has a “long pointy bill” which is the counterfactual attribute for *Least Auklet*. On the other hand, a *Cardinal* is classified as a cardinal because of the “red bird” and “black face” attributes while not as a *Scarlet Tanager* because of the lack of “black wings”. These results show that our counterfactual explanations do not always generate the same phrases for the counterfactual classes. Our counterfactual explanations talk about properties of the counterfactual class that are not relevant to the particular query image, whose evidence is clearly visible in both the counterfactual and the query images.



The AI justified its prediction with the following evidence: this is a small orange bird with a black wing and a small black beak. Do you think you would accept the Als prediction?

○ Accept prediction
○ Do not accept prediction

The AI justified its prediction with the following evidence: this is a brown and black spotted bird with a white belly. Do you think you would accept the Als prediction?

○ Accept prediction
○ Do not accept prediction

AI Correct: User should accept prediction

AI Incorrect: User should accept prediction

Figure 5.10: Example UI for our task which tests explanation usefulness. Users are asked whether or not they should accept a decision from an AI given the explanation provided by the AI system.

In conclusion, counterfactual explanations go one step further in language-based explanation generation. Contrasting a class with another closely related class helps the user build a more coherent cognitive representation of a particular object of interest.

5.4.4.4 Are Explanations Helpful to Humans?

As a final evaluation, we test if our explanations can be helpful to humans to understand whether or not to accept an AI decision. In order to conduct this test, we ask humans to first go through a training stage where they are provided example images of birds, the ground truth and predicted class, as well as an explanation. After the training stage, they are asked to look at explanatory text and decide whether or not they would accept the AI decision given the explanatory text. Note that we do not include the predicted class (e.g., “The model predicted that this is a cardinal”) because different human users might have different knowledge of different bird varieties. If a human is a more experienced bird watcher, they could determine if they should reject the decision based only on the label. Thus to rid our experiment of this bias, humans cannot see the predicted class label. Figure 5.10 shows an example of our user interface for an example where the human should trust the AI decision (left) and where the human should not trust the AI decision (right). As a baseline, we consider how well a human can determine whether or not to accept an AI decision given only the image.

Figure 5.11 shows our results. 50% of the examples humans see correspond to instances where the predicted label is *incorrect*. Thus a random baseline is 50%. Without an explanation, humans perform close to chance, but with our explanations human performance increased to roughly 62%. Results are significant with a p-value < 0.01. As mentioned earlier, our experiments were based off experiments in [Cha+18] which showed that visual explanations were not helpful for a similar task. We believe textual explanations are particularly helpful because they are explicit and easy to understand for non-experts.

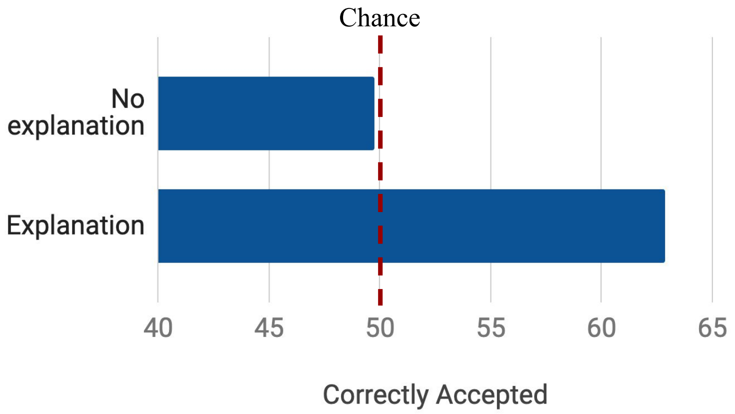


Figure 5.11: Results on our usefulness task. When provided with our explanations, humans are better able to determine whether or not they should accept or reject an AI decision.

5.5 Discussion

In this chapter we have discussed how to generate textual explanations which are both class discriminative and image relevant. In order to ensure explanations are class discriminative, we proposed a novel explanation sampler trained with a discriminative loss. To ensure image relevance, we proposed the phrase critic model which selects the best explanation given a set of explanations by grounding explanatory evidence in an image. We further demonstrated the ability of the phrase critic by considering the FOIL tasks. We also showed that our phrase critic pipeline could help us build *counterfactual* explanations which can discuss how visual evidence might change a classifier decision. Finally, we demonstrated that explanations are helpful to humans when humans are asked to decide whether to accept or reject a decision.

We have focussed on generating textual explanations which are easy for humans to understand and interpret. However, other explanation modalities such as visual explanations which highlight important parts of an explanation [Ram+17; Sel+17], exemplar explanations which show similar examples [Che+18a], or neuron activation explanations which show which neurons were activated for a decision [Bau+17] provide complementary information. Though some work has considered combining textual and visual explanations [HP+18; WM18], understanding which explanations are most useful and how to integrate different explanation modalities in a helpful way is ongoing work.

One assumption we make in our current system is that all humans will be satisfied by the same explanation. However, a bird expert might expect a very different explanation for a bird species than a novice bird watcher. Explanations between humans are inherently social, with the explainer and explainee engaging in discussion and interaction relative to each other's beliefs [Mil18]. Modelling the current beliefs of a human could help an explanation system decide which evidence to show next, or what kind of explanation would be most helpful to a human (e.g., is it better to give a text explanation or a saliency based explanation?). Some work considers explanations for teaching a task to humans (e.g., fine-grained recognition [MA+18]). In these domains, modeling the belief of a student could lead to more beneficial explanations.

One open question in the explanation space is whether generated explanations can actually improve a classification decision. In the above work, we mainly focussed on explaining decisions in a post-hoc fashion: in other words, our explanation mechanism did not impact the classification decision. In prior work, interpretable systems which require changing the original model architecture have sometimes led to a decrease in model accuracy [AA19; ZNWZ18]. However, humans are capable of both making decisions and explaining them so this may not be a necessary trade-off. In fact, humans actually exploit explanations in their learning process [Wal+17; Chi+94; WL13]. Our phrase critic classifier (which we use to evaluate the class discriminativeness of generated text) could be considered an interpretable classifier. As we change different input features (for example, change the explanation text from "brown wing" to "white wing") we can observe that the classification score will also change. Thus, the phrase critic classifier is semantically interpretable. However, the maximum accuracy we observe using our phrase critic classifier is 65.6%, substantially lower than the accuracy of the original MCB classifier (83.9%). In the future including the generated explanation learning process, and showing that it can even improve the classification accuracy, is an exciting direction.

Chapter 6

Localizing Moments in Video with Natural Language

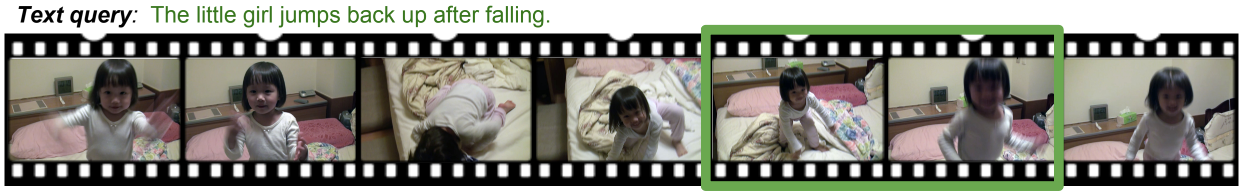


Figure 6.1: We propose the task of moment localization with natural language. The input is a longer video and a natural language expression, and the output is the start and end point which corresponds to *when* the natural language query occurs in the video

Most work in this thesis has thus far focussed on one task at the intersection of language and vision: generating natural language expressions given a visual input. However, a variety of other tasks require a joint understanding of language and vision. For example, consider the video depicted in Figure 6.1, in which a little girl jumps around, falls down, and then gets back up to start jumping again. Suppose we want to refer to a particular temporal segment, or moment, from the video, such as when the girl resiliently begins jumping again after she has fallen. Simply referring to the moment via an action, object, or attribute keyword may not uniquely identify it. For example, important objects in the scene, such as the girl, are present in each frame. Likewise, recognizing all the frames in which the girl is jumping will not localize the moment of interest as the girl jumps both before and after she has fallen. Rather than being defined by a single object or activity, the moment may be defined by when and how specific actions take place *in relation* to other actions. An intuitive way to refer to the moment is via a natural language phrase, such as "the little girl jumps back up after falling". [1]

[1] This chapter is based on joint work done with Oliver Wang, Eli Shechtman, Josef Sivic, Trevor Darrell, and Bryan Russell [Hen+17; Hen+18b] presented at ICCV 2017 and EMNLP 2018.

Motivated by this example, we consider localizing moments in video with natural language. Specifically, given a video and text description, we identify start and end points in the video which correspond to the given text description. This is a challenging task requiring both language and video understanding, with important applications in video retrieval, such as finding particular moments from a long personal holiday video, or desired B-roll stock video footage from a large video library (e.g., Adobe Stock[2], Getty[3], Shutterstock[4]).

Existing methods for natural language based video retrieval [Ota+16; Xu+15b; TTS16] retrieve an entire video given a text string but do not identify *when* a moment occurs within a video. To localize moments within a video we propose to learn a joint video-language model in which referring expressions and video features from corresponding moments are close in a shared embedding space. However, in contrast to whole video retrieval, we argue that in addition to video features from a specific moment, global video context and knowing when a moment occurs within a longer video are important cues for moment retrieval. For example, consider the text query "The man on the stage comes closest to the audience". The term "closest" is relative and requires temporal context to properly comprehend. Additionally, the temporal position of a moment in a longer video can help localize the moment. For the text query "The biker starts the race", we expect moments earlier in the video in which the biker is racing to be closer to the text query than moments at the end of the video. We thus propose the Moment Context Network (MCN) which includes a global video feature to provide temporal context and a temporal endpoint feature to indicate when a moment occurs in a video.

A major obstacle when training our model is that current video-language datasets do not include natural language which can uniquely localize a moment. Additionally, datasets like [Lin+14a; Reg+13] are small and restricted to specific domains, such as dash-cam or cooking videos, while datasets [CD11; Roh+17a; Xu+16] sourced from movies and YouTube are frequently edited and tend to only include entertaining moments (see [Sig+16] for discussion). We believe the task of localizing moments with natural language is particularly interesting in unedited videos which tend to include uneventful video segments that would generally be cut from edited videos. Consequently, we desire a dataset which consists of distinct moments from unedited video footage paired with descriptions which can uniquely localize each moment, analogous to datasets that pair distinct image regions with descriptions [Kaz+14; Mao+16].

To address this problem, we collect the Distinct Describable Moments (DiDeMo) dataset which includes distinct video moments paired with descriptions which uniquely localize the moment in the video. Our dataset consists of over 10,000 unedited videos with 3-5 pairs of descriptions and distinct moments per video. DiDeMo is collected in an open-world setting and includes diverse content such as pets, concerts, and sports games. To ensure that descriptions are referring and thus uniquely localize a moment, we include a validation step inspired by [Kaz+14].

[2] https://stock.adobe.com

[3] http://www.gettyimages.com

[4] https://www.shutterstock.com

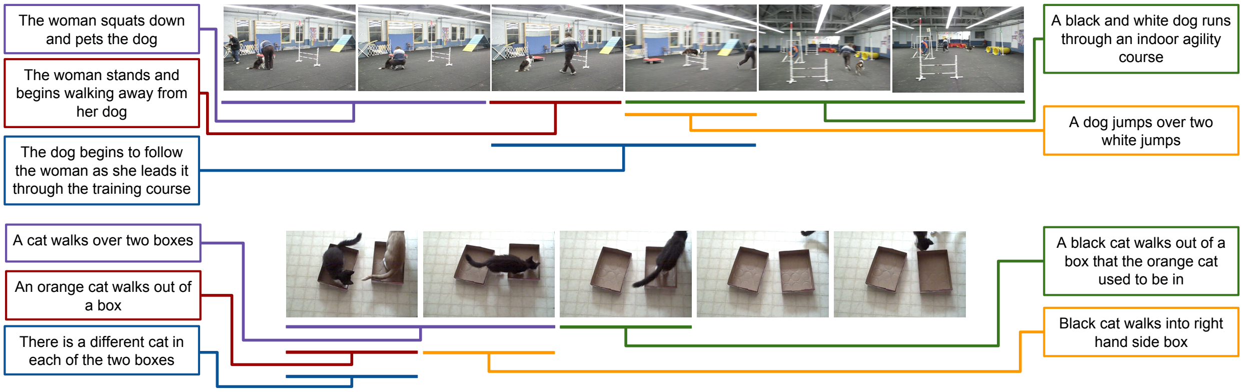

The woman squats down and pets the dog
The woman stands and begins walking away from her dog
The dog begins to follow the woman as she leads it through the training course
A black and white dog runs through an indoor agility course
A dog jumps over two white jumps
A cat walks over two boxes
An orange cat walks out of a box
There is a different cat in each of the two boxes
A black cat walks out of a box that the orange cat used to be in
Black cat walks into right hand side box

Figure 6.2: Example videos and annotations from our Distinct Describable Moments (DiDeMo) dataset. Annotators describe moments with varied language (e.g., "A cat walks over two boxes" and "An orange cat walks out of a box"). Videos with multiple events (top) have annotations which span all five-second segments. Other videos have segments in which no distinct event takes place (e.g., the end of the bottom video in which no cats are moving).

6.1 New Data for a New Task: Distinct Describable Moments Dataset

A major challenge when designing algorithms to localize moments with natural language is that there is a dearth of large-scale datasets which consist of referring expressions and localized video moments. To mitigate this issue, we introduce the Distinct Describable Moments (DiDeMo) dataset which includes over 10,000 25-30 second long personal videos with over 40,000 localized text descriptions. Example annotations are shown in Figure 6.2.

6.1.1 Dataset Collection

To ensure that each description is paired with a single distinct moment, we collect our dataset in two phases (similar to how [Kaz+14] collected text to localize image regions). First, we asked annotators to watch a video, select a moment, and describe the moment such that another user would select the same moment based on the description. Then, descriptions collected in the first phase are validated by asking annotators to watch videos and mark moments that correspond to collected descriptions.

Harvesting Personal Videos. We randomly select over 14,000 videos from YFCC100M [Tho+15] which contains over 100,000 Flickr videos with a Creative Commons License. To ensure harvested videos are unedited, we run each video through a shot detector based on the difference of color histograms in adjacent frames [MF03] then manually filter videos which are not caught. Videos in

Dataset	# Videos/ # Clips	# Sentences	Video Source	Domain	Temporal Localization	Un-Edited	Referring Expressions
YouCook [Das+13]	88/-	2,668	YouTube	Cooking			
Charades [Sig+16]	10,000/-	16,129	Homes	Daily activities		✓	
TGIF [Li+16]	100,000 /-	125,781	Tumblr GIFs	Open			
MSVD [CD11]	1,970/1,970	70,028	YouTube	Open	✓		
MSR-VTT [Xu+16]	7,180/10,000	200,000	YouTube	Open	✓		
LSMDC 16 [Roh+17a]	200/128,085	128,085	Movie	Open	✓		
TV Dataset [YFFF14]	4/1,034	1,034	TV Shows	TV Shows	✓		
KITTI [Lin+14a]	21/520	520	Car Camera	Driving	✓	✓	
TACoS [Reg+13; Roh+13]	123/7,206	18,227	Lab Kitchen	Cooking	✓	✓	
TACoS multi-level[Roh+14]	185/14,105	52,593	Lab Kitchen	Cooking	✓	✓	
UT Egocentric [YFFF14]	4/11,216	11,216	Egocentric	Daily Activities	✓	✓	
Disneyland [YFFF14]	8/14,926	14,916	Egocentric	Disneyland	✓	✓	
DiDeMo	10,464/26,892	40,543	Flickr	Open	✓	✓	✓

Table 6.1: Comparison of DiDeMo to other video-language datasets. DiDeMo is unique because it includes a validation step ensuring that descriptions are referring expressions.

DiDeMo represent a diverse set of real-world videos, which include interesting, distinct moments, as well as uneventful segments which might be excluded from edited videos.

Video Interface. Localizing text annotations in video is difficult because the task can be ambiguous and users must digest a 25-30s video before scrubbing through the video to mark start and end points. To illustrate the inherent ambiguity of our task, consider the phrase "The woman leaves the room." Some annotators may believe this moment begins as soon as the woman turns towards the exit, whereas others may believe the moment starts as the woman's foot first crosses the door threshold. Both annotations are valid, but result in large discrepancies between start and end points.

To make our task less ambiguous and speed up annotation, we develop a user interface in which videos are presented as a timeline of temporal segments. Each segment is displayed as a gif, which plays at 2x speed when the mouse is hovered over it. Following [YFFF14], who collected localized text annotations for summarization datasets, we segment our videos into 5-second segments. Users select a moment by clicking on all segments which contain the moment. To validate our interface, we ask five users to localize moments in ten videos using our tool and a traditional video scrubbing tool. Annotations with our gif-based tool are faster to collect (25.66s vs. 38.48s). Additionally, start and end points marked using the two different tools are similar. The standard deviation for start and end points marked when using the video scrubbing tool (2.49s) is larger than the average difference in start and end points marked using the two different tools (2.45s).

Moment Validation. After annotators describe a moment, we ask three additional annotators to localize the moment given the text annotation and the same video. To accept a moment description, we require that at least three out of four annotators (one describer and three validators) be in agreement. We consider two annotators to agree if one of the start *or* end point differs by at most one gif.

6.1.2 DiDeMo Summary

Table 6.1 compares our Distinct Describable Moments (DiDeMo) dataset to other video-language datasets. Though some datasets include temporal localization of natural language, these datasets do not include a verification step to ensure that descriptions refer to a single moment. In contrast, our verification step ensuring that descriptions in DiDeMo are *referring expressions*, meaning that they refer to a specific moment in a video.

Vocabulary. Because videos are curated from Flickr, DiDeMo reflects the type of content people are interested in recording and sharing. Consequently, DiDeMo is human-centric with words like "baby", "woman", and "man" appearing frequently. Since videos are randomly sampled, DiDeMo has a long tail with words like "parachute" and "violin", appearing infrequently (28 and 38 times).

Important, distinct moments in a video often coincide with specific camera movements. For example, "the camera pans to a group of friends" or "zooms in on the baby" can describe distinct moments. Many moments in personal videos are easiest to describe in reference to the viewer (e.g., "the little boy runs towards the camera"). In contrast to other dataset collection efforts [CD11], we allow annotations to reference the camera, and believe such annotations may be helpful for applications like text-assisted video editing.

Table 6.2 contrasts the kinds of words used in DiDeMo to two natural language object retrieval datasets [Kaz+14; Mao+16] and two video description datasets [Roh+17a; Xu+16]. The three left columns report the percentage of sentences which include camera words (e.g., "zoom", "pan", "cameraman"), temporal indicators (e.g., "after" and "first"), and spatial indicators (e.g., "left" and "bottom"). We also compare how many words belong to certain parts of speech (verb, noun, and adjective) using the natural language toolkit part-of-speech tagger [BKL09]. DiDeMo contains more sentences with temporal indicators than natural language object retrieval and video description datasets, as well as a large number of spatial indicators. DiDeMo has a higher percentage of verbs than natural language object retrieval datasets, suggesting understanding action is important for moment localization in video.

Annotated Time Points. Annotated segments can be any contiguous set of gifs. Annotators generally describe short moments with 72.34% of descriptions corresponding to a single gif and 22.26% corresponding to two contiguous gifs. More annotated moments occur at the beginning of a video than the end. This is unsurprising as people generally choose to begin filming a video when something interesting is about to happen. In 86% of videos annotators described multiple distinct moments with an average of 2.57 distinct moments per video.

	% Sentences			% Words		
	Camera	Temp.	Spatial	Verbs	Nouns	Adj.
ReferIt [Kaz+14]	0.33	1.64	43.13	5.88	52.38	11.54
RefExp [Mao+16]	1.88	1.00	15.11	8.97	36.26	11.82
MSR-VTT [Xu+16]	2.10	2.03	1.24	18.77	36.95	5.12
LSMDC 16 [Roh+17a]	1.09	7.58	1.49	13.71	37.44	3.99
DiDeMo	19.69	18.42	11.62	16.06	35.26	7.89

Table 6.2: DiDeMo contains more camera and temporal words than natural language object recognition datasets [Kaz+14; Mao+16] or video description datasets [Xu+16; Roh+17a]. Additionally, verbs are more common in DiDeMo than in natural language object retrieval datasets suggesting natural language moment retrieval relies more heavily on recognizing actions than natural language object retrieval.

6.2 Model: Moment Context Network

Our moment retrieval model effectively localizes natural language queries in longer videos. Given input video frames $v = \{v_t\}$, where $t \in \{0, \ldots, T-1\}$ indexes time, and a proposed temporal interval, $\hat{\tau} = \tau_{start} : \tau_{end}$, we extract visual temporal context features which encode the video moment by integrating both local features and global video context. Given a sentence s we extract language features using an LSTM [HS97] network. At test time our model optimizes the following objective

$$\hat{\tau} = \operatorname*{argmin}_{\tau} D_\theta(s, v, \tau), \tag{6.1}$$

where D_θ is a joint model over the sentence s, video v, and temporal interval τ given model parameters θ (Figure 6.3).

Visual Temporal Context Features. We encode video moments into visual temporal context features by integrating local video features, which reflect what occurs within a specific moment, global video features, which provide context for a video moment, and temporal endpoint features, which indicate when a moment occurs within a longer video. To construct local and global video features, we first extract high level video features using a deep convolutional network for each video frame, then average pool video features across a specific time span (similar to features employed by [Ven+15b] for video description and [TTS16] for whole video retrieval). Local features are constructed by pooling features within a specific moment and global features are constructed by averaging over all frames in a video.

When a moment occurs in a video can indicate whether or not a moment matches a specific query. To illustrate, consider the query “the bikers start the race.” We expect moments closer to the beginning of a video in which bikers are racing to be more similar to the description than moments at the end of the video in which bikers are racing. To encode this temporal information, we include temporal endpoint features which indicate the start and endpoint of a candidate moment (normalized to the interval $[0, 1]$). We note that our global video features and temporal endpoint

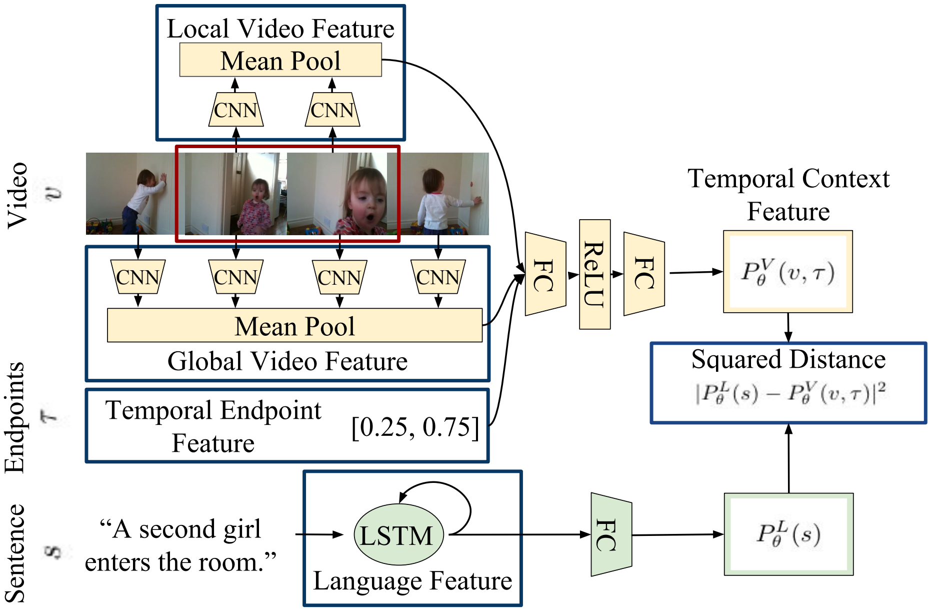


Figure 6.3: Our Moment Context Network (MCN) learns a shared embedding for video temporal context features and LSTM language features. Our video temporal context features integrate local video features, which reflect what occurs during a specific moment, global features, which provide context for the specific moment, and temporal endpoint features which indicate when a moment occurs in a video. We consider both appearance and optical flow input modalities, but for simplicity only show the appearance input modality here.

features are analogous to global image features and spatial context features frequently used in natural language object retrieval [Hu+16; Mao+16].

Localizing video moments often requires localizing specific activities (like "jump" or "run"). Therefore, we explore two sources of visual input modalities; appearance or RGB frames (v_t) and optical flow frames (f_t). We extract fc_7 features from RGB frames using VGG [SZ15] pre-trained on ImageNet [Rus+15]. We expect these features to accurately identify specific objects and attributes in video frames. Likewise, we extract optical flow features from the penultimate layer from a competitive activity recognition model [Wan+16]. We expect these features to help localize moments which require understanding action.

Temporal context features are extracted by inputting local video features, global video features, and temporal endpoint features into a two layer neural network with ReLU nonlinearities (Figure 6.3 top). Separate weights are learned when extracting temporal context features for RGB frames (denoted as P_θ^V) and optical flow frames (denoted as P_θ^F).

Language Features. To capture language structure, we extract language features using a recurrent network (specifically an LSTM [HS97]). After encoding a sentence with an LSTM, we pass the last hidden state of the LSTM through a single fully-connected layer to yield embedded feature

P^L_θ. Though our dataset contains over 40,000 sentences, it is still small in comparison to datasets used for natural language object retrieval (e.g., [Kaz+14; Mao+16]). Therefore, we find that representing words with dense word embeddings (specifically Glove [PSM14]) as opposed to one-hot encodings yields superior results when training our LSTM.

Joint Video and Language Model. Our joint model is the sum of squared distances between embedded appearance, flow, and language features

$$D_\theta(s, v, \tau) = |P^V_\theta(v, \tau) - P^L_\theta(s)|^2 + \eta|P^F_\theta(f, \tau) - P^L_\theta(s)|^2, \tag{6.2}$$

where η is a tunable (via cross validation) "late fusion" scalar parameter. η was set to 2.33 via ablation studies.

Ranking Loss for Moment Retrieval. We train our model with a ranking loss which encourages referring expressions to be closer to corresponding moments than negative moments in a shared embedding space. Negative moments used during training can either come from different segments within the same video (intra-video negative moments) or from different videos (inter-video negative moments). Revisiting the video depicted in Figure **??**, given a phrase "the little girl jumps back up after falling" many intra-video negative moments include concepts mentioned in the phrase such as "little girl" or "jumps". Consequently, our model must learn to distinguish between subtle differences within a video. By comparing the positive moment to the intra-video negative moments, our model can learn that localizing the moment corresponding to "the little girl jumps back up after falling" requires more than just recognizing an object (the girl) or an action (jumps). For training example i with endpoints τ_i, we define the following intra-video ranking loss

$$\mathcal{L}^{intra}_i(\theta) = \sum_{n \in \Gamma \setminus \tau^i} \mathcal{L}^R\left(D_\theta(s^i, v^i, \tau^i), D_\theta(s^i, v^i, n)\right), \tag{6.3}$$

where $\mathcal{L}^R(x, y) = \max(0, x - y + b)$ is the ranking loss, Γ are all possible temporal video intervals, and b is a margin. Intuitively, this loss encourages text queries to be closer to a corresponding video moment than all other possible moments from the same video.

Only comparing moments within a single video means the model must learn to differentiate between subtle differences without learning how to differentiate between broader semantic concepts (e.g., "girl" vs. "sofa"). Hence, we also compare positive moments to inter-video negative moments which generally include substantially different semantic content. When selecting inter-video negative moments, we choose negative moments which have the same start and end points as positive moments. This encourages the model to differentiate between moments based on semantic content, as opposed to when the moment occurs in the video. During training we do not verify that inter-video negatives are indeed true negatives. However, the language in our dataset is diverse enough that, in practice, we observe that randomly sampled inter-video negatives are generally true negatives. For training example i, we define the following inter-video ranking loss

$$\mathcal{L}^{inter}_i(\theta) = \sum_{j \neq i} \mathcal{L}^R\left(D_\theta(s^i, v^i, \tau^i), D_\theta(s^i, v^j, \tau^i)\right). \tag{6.4}$$

	Baseline Comparison (Test Set)			
	Model	Rank@1	Rank@5	mIoU
1	Upper Bound	74.75	100.00	96.05
2	Chance	3.75	22.50	22.64
3	Moment Frequency Prior	19.40	66.38	26.65
4	CCA	18.11	52.11	37.82
5	Natural Lang. Obj. Retrieval [Hu+16]	16.20	43.94	27.18
6	Natural Lang. Obj. Retrieval [Hu+16] (re-trained)	15.57	48.32	30.55
7	MCN (ours)	**28.10**	**78.21**	**41.08**
	Ablations (Validation Set)			
8	LSTM-RGB-local	13.10	44.82	25.13
9	LSTM-Flow-local	18.35	56.25	31.46
10	LSTM-Fusion-local	18.71	57.47	32.32
11	LSTM-Fusion + global	19.88	62.39	33.51
12	LSTM-Fusion + global + tef (MCN)	**27.57**	**79.69**	**41.70**

Table 6.3: Our Moment Context Network (MCN) outperforms baselines (rows 1-6) on our test set. We show ablation studies on our validation set in rows 8-12. Both flow and RGB modalities are important for good performance (rows 8-10). Global video features and temporal endpoint features (tef) both lead to better performance (rows 10-12).

This loss encourages text queries to be closer to corresponding video moments than moments outside the video, and should thus learn to differentiate between broad semantic concepts. Our final inter-intra video ranking loss is

$$\mathcal{L}(\theta) = \lambda \sum_i \mathcal{L}_i^{intra}(\theta) + (1-\lambda) \sum_i \mathcal{L}_i^{inter}(\theta), \tag{6.5}$$

where λ is a weighting parameter chosen through cross-validation.

6.3 Results: Moment Context Network

In this section we report qualitative and quantitative results on DiDeMo. First, we describe our evaluation criteria and then evaluate against baseline methods.

Metrics: Accounting for Human Variance. Our model ranks candidate moments in a video based on how well they match a text description. Candidate moments come from the temporal segments defined by the gifs used to collect annotations. A 30 second video will be broken into six five-second gifs. Moments can include any contiguous set of gifs, so a 30-second video contains 21 possible moments. We measure the performance of each model with Rank@1 (R@1), Rank@5

(R@5), and mean intersection over union (mIoU). Instead of consolidating all human annotations into one ground truth, we compute the score for a prediction and each human annotation for a particular description/moment pair. To account for outlier annotations, we consider the highest score among sets of annotations A' where A' are the four-choose-three combinations of all four annotations A. Hence, our final score for a prediction P and four human annotations A using metric M is: $score(P, A) = \max_{A' \in \binom{A}{3}} \frac{1}{3} \sum_{a \in A'} M(P, a)$. As not all annotators agree on start and end points it is impossible to achieve 100% on all metrics (c.f., upper bounds in Table 6.3).

Baseline: Moment Frequency Prior. Though annotators may mark any contiguous set of gifs as a moment, they tend to select short moments toward the beginning of videos. The moment frequency prior selects moments which correspond to gifs most frequently described by annotators.

Baseline: CCA. Canonical correlation analysis (CCA) achieves competitive results for both natural language image [Kle+15] and object [Plu+15] retrieval tasks. We use the CCA model of [Kle+15] and employ the same visual features as the MCN model. We extract language features from our best MCN language encoder for fair comparison.

Baseline: Natural Language Object Retrieval. Natural language object retrieval models localize objects in a text image. We verify that localizing objects is not sufficient for moment retrieval by running a natural language object retrieval model [Hu+16] on videos in our test set. For every tenth frame in a video, we score candidate bounding boxes with the object retrieval model proposed in [Hu+16] and compute the score for a frame as the maximum score of all bounding boxes. The score for each candidate moment is the average of scores for frames within the moment. Additionally, we re-train [Hu+16] using the same feautures used to train our MCN model; instead of candidate bounding boxes, we provide candidate temporal chunks and train with both appearance and flow input modalities.

Implementation Details. DiDeMo videos are split into training (8,395), validation (1,065), and testing (1,004) sets. Videos from a specific Flickr user only appear in one set. All models are implemented in Caffe [Jia+14] and have been publicly released [5]. SGD (mini-batch size of 120) is used for optimization and all hyperparamters, such as embedding size (100), margin (0.1), and LSTM hidden state size (1000), are chosen through ablation studies.

6.3.1 Results

Table 6.3 compares different variants of our proposed retrieval model to our baselines. Our ablations demonstrate the importance of our temporal context features and the need for both appearance and optical flow features.

[5] https://people.eecs.berkeley.edu/~lisa_anne/didemo.html

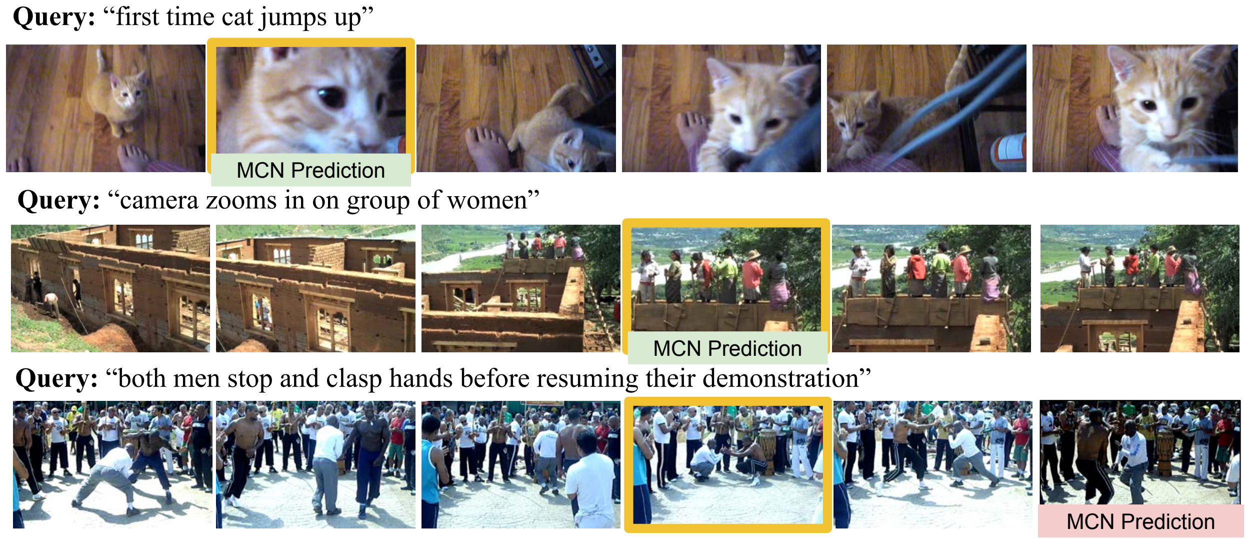

Query: "first time cat jumps up"
MCN Prediction
Query: "camera zooms in on group of women"
MCN Prediction
Query: "both men stop and clasp hands before resuming their demonstration"
MCN Prediction

Figure 6.4: Natural language moment retrieval results on DiDeMo. Ground truth moments are outlined in yellow. The Moment Context Network (MCN) localizes diverse descriptions which include temporal indicators, such as "first" (top), and camera words, such as "camera zooms" (middle).

Baseline Comparison. Rows 1-7 of Table 6.3 compare the Moment Context Network (MCN) model to baselines on our test set. Though all baselines we trained (lines 4-6) have similar R@1 and R@5 performance, CCA performs substantially better on the mIoU metric. Scoring video segments based on the scores from a natural language object retrieval model [Hu+16] does fairly well, performing similarly to the same model retrained with our features. This suggests that pretraining with a dataset designed for natural language object retrieval and incorporating spatial localization into our model could improve results. We believe that retraining [Hu+16] leads to poor results on our dataset because it relies on sentence generation rather than directly retrieving a moment. Additionally, our model does substantially better than the moment frequency prior.

Visual Temporal Context Feature. Rows 9-12 of Table 6.3 demonstrate the importance of temporal context for moment retrieval. The inclusion of both the global video feature and temporal endpoint feature increase performance considerably. Additionally, we find that combining both appearance and optical flow features is important for best performance.

Qualitative Results. Figure 6.4 shows moments predicted by MCN. Our model is capable of localizing a diverse set of moments including moments which require understanding temporal indicators like "first" (Figure 6.4 top) as well as moments which include camera motion (Figure 6.4 middle).

Fine-grained Moment Localization Even though our ground truth moments correspond to five-second chunks, we can evaluate our model on smaller temporal segments at test time to predict

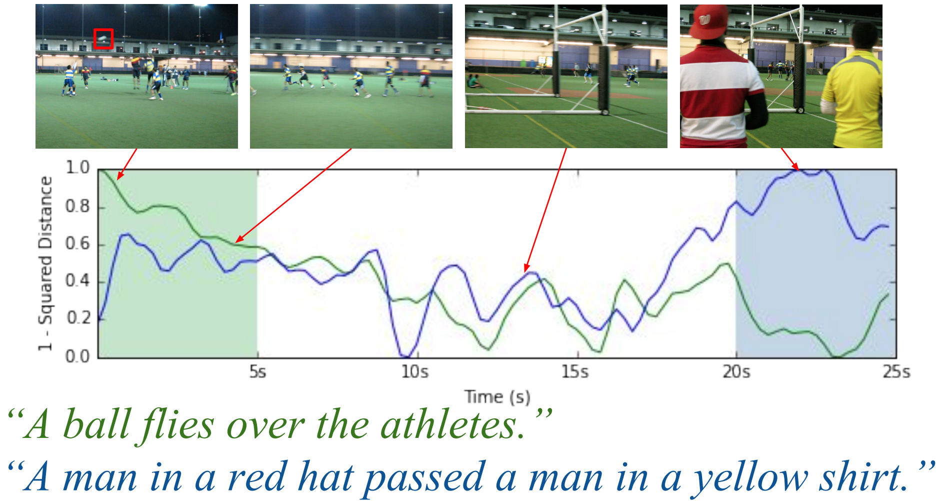


Figure 6.5: MCN correctly retrieves two different moments (light green rectangle on left and light blue rectangle on right). Though our ground truth annotations are five-second segments, we can evaluate with more fine-grained temporal proposals at test time. This gives a better understanding of when moments occur in video (e.g., "A ball flies over the athletes" occurs at the start of the first temporal segment).

moment locations with finer granularity. Instead of extracting features for a five second segment, we evaluate on individual frames extracted at $\sim$ 3 fps. Figure 6.5 includes an example in which two text queries ("A ball flies over the athletes" and "A man in a red hat passed a man in a yellow shirt") are correctly localized by our model. The frames which best correspond to "A ball flies over the athletes" occur in the first few seconds of the video and the moment "A man in a red hat passed a men in a yellow shirt" finishes before the end point of the fifth segment.

Limitations. One limitation of the current MCN model is that it cannot explicitly model temporal relationships between actions. For example, a global video feature can encode important video context, but is not helpful for localizing a moment like "the little girl jumps after falling" in which the relationship between separate actions like "jumps" and "falling" must be understood. Though queries like this do exist in the DiDeMo dataset (e.g., Figure 6.1), they are not common enough to reliably train and test models which can consistently localize moments which use temporal language. A similar shortcoming has been observed in other language and vision tasks. For example, many current VQA systems are not capable of answering questions with complex compositional queries (e.g., "What color is the dog on the sofa to the left of the white cat?"). In order to solve this issue, researchers have pursued not only better models [And+16c; Hu+17a; Joh+17b; San+17], but also better datasets which require a model to learn more complex reasoning [Joh+17a; HM19]. Following this trend, we next consider a complementary dataset to the DiDeMo dataset, called TEMPOral Reasoning in Video and Language (TEMPO), which explicitly requires temporal reasoning in order to localize video moments. Using this dataset we will propose a new model, the Moment Localization with Latent Context (MLLC) model, which explicitly reasons about temporal relationships between different video moments to better localize queries which require temporal

Dataset	Before	After	Then	While	Yet	During	Until
TACoS	50	62	731	82	23	0	4
Charades-STA	281	27	1873	1165	0	3	1
DiDeMo	198	119	1021	266	16	21	22
TEMPO - TL	23,842	23842	11921	-	-	-	-
TEMPO - HL	6610	5495	5478	5425	-	-	-

Table 6.4: Word frequency of temporal words in natural language moment localization datasets.

reasoning.

6.4 TEMPOral reasoning in Video and Language

The TEMPOral reasoning in video and language (TEMPO) dataset is explicitly designed to test a model's ability to localize video moments which require understanding temporal relationships between actions. Our dataset consists of two parts: TEMPO - Template Language (TL) and TEMPO - Human Language (HL). We create TEMPO - TL using language templates to augment the original sentences in DiDeMo with temporal words. The template allows us to generate a large number of sentences with known ground truth base and context moments. However, template language lacks the complexity of human language, so we then collect an additional fully user-constructed dataset, TEMPO - HL, consisting of sentences that contain specific temporal words.

Temporal Words in Current Datasets. We first analyze temporal words which occur in current natural language moment retrieval datasets, including TACoS [Reg+13] which is a smaller dataset restricted to the cooking domain, Charades-STA [Gao+17] which was concurrently released with DiDeMo, and our DiDeMo dataset. We consider temporal adjectives, adverbs, and prepositions found both by closely analyzing moment-localization datasets and consulting lists containing words which belong to different parts of speech. In particular, we rely on the preposition project [LH05][6] to scrape relevant temporal words. Table 6.4 shows example temporal words and the number of times they occur in each dataset. Though all moment localization datasets use temporal words, they do not contain enough examples to reliably train and evaluate current models. Additionally, we observe that temporal words which are frequently used when describing video segments are different than those commonly used in text without video grounding. For example, in [PH04], "during" is a common example, but we observe that "during" is infrequently used when describing video. Of temporal words, we focus on the four most common words, "before", "after", "then", and "while" when creating our dataset.

TEMPO - Template Language. To construct sentences in TEMPO-TL, we find adjacent moments in the DiDeMo dataset and fill in template sentences for "before", "after", and "then" temporal words. For "before", we use two templates: "X before Y" and "Before Y, X", where

[6] http://www.clres.com/prepositions.html

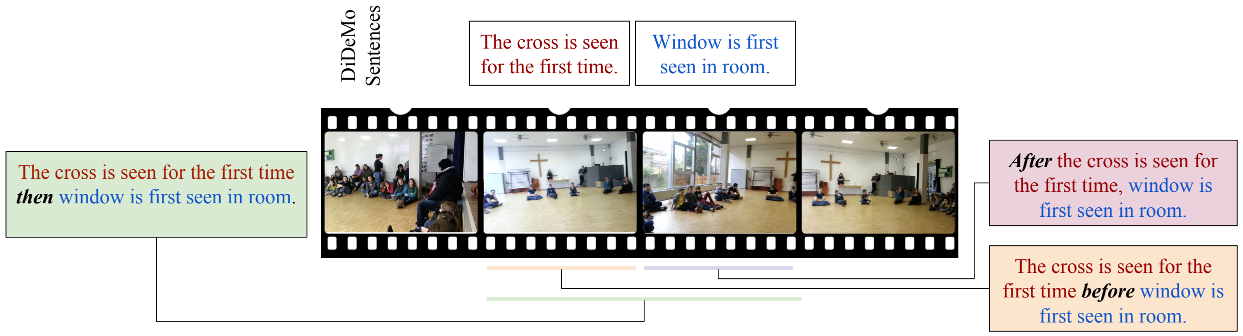

DiDeMo Sentences
The cross is seen for the first time.
Window is first seen in room.
The cross is seen for the first time then window is first seen in room.
After the cross is seen for the first time, window is first seen in room.
The cross is seen for the first time before window is first seen in room.

Figure 6.6: Example sentences in TEMPO - TL. The top sentence correspond to original DiDeMo sentences. TEMPO-TL sentences are created by merging different DiDeMo sentences with temporal words like "before", "after", and "then".

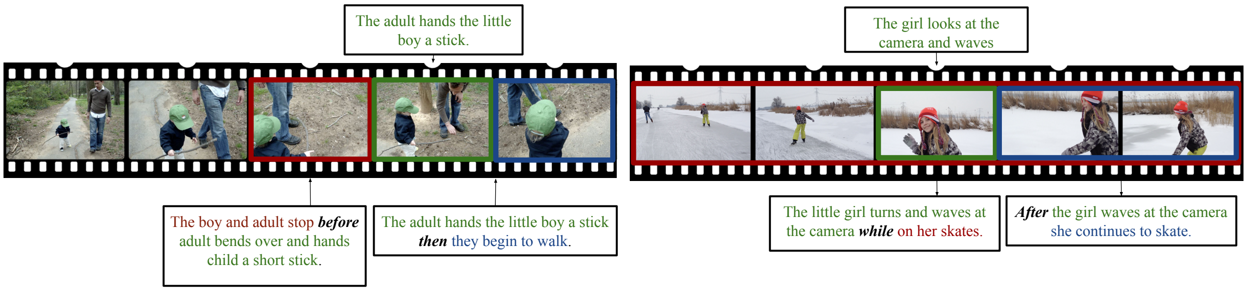

The adult hands the little boy a stick.
The boy and adult stop before adult bends over and hands child a short stick.
The adult hands the little boy a stick then they begin to walk.
The girl looks at the camera and waves
The little girl turns and waves at the camera while on her skates.
After the girl waves at the camera she continues to skate.

Figure 6.7: Example sentences in TEMPO - HL. The top sentence corresponds to the reference moment (shown in green). The bottom sentences are newly collected sentences which use temporal language.

X and Y are sentences from the original DiDeMo dataset. Likewise for "after", we consider the templates "X after Y" and "After Y, X". For "then" we only consider one template, "X then Y." Figure 6.6 shows an example of our TEMPO-TL annotations.

TEMPO - Human Language. Though the template dataset is an interesting testbed for understanding temporal language, it is difficult to replicate the interesting complexities in human language. For example, when writing long sentences with temporal prepositions, humans frequently make use of language structure such as coreference to form more cohesive statements.

To collect annotations, we follow the protocol in [Hen+17] and segment videos into 5-second temporal segments. After collecting descriptions, we ensure descriptions are localizable by asking other workers to localize each moment. To collect data for "before", "after", and "then", we ask annotators to describe a segment in *relation* to a "reference" moment from the DiDeMo dataset. For example, if the DiDeMo dataset includes a localized phrase like "the cat jumps", annotators write a sentence which refers to the segment "the cat jumps" using a specific temporal word. We provide both the phrase ("the cat jumps") and the reference moment to annotators, and the annotators provide a sentence describing a new moment which references the reference moment.

TEMPO-HL includes unique properties which are hard to replicate with template data. Figure 6.7 depicts the base moment provided to workers, as well as descriptions from TEMPO-HL. In Figure 6.7, the description "The adult hands the little boy the stick then they walk away" includes an example of visual coreference ("they"). We note that use of pronouns is much more prevalent in TEMPO-HL, with 28.1% of sentences in TEMPO-HL including pronouns ("he", "she", "it") in contrast to 10.3% of sentences in the original DiDeMo dataset. Additionally, annotators will refer to the base moment with different language than originally used in the base moment (e.g., "the girl waves at the camera" versus the base moment "the girl looks at the camera and waves") in order to make their sentences more fluent.

6.5 Model: Moment Localization with Latent Context

Given a video v and natural-language query q describing a moment in the video, our goal is to output the moment $\tau = \left(\tau^{(s)}, \tau^{(e)}\right)$ where $\tau^{(s)}$ and $\tau^{(e)}$ are temporal start and end points in the video, respectively. In the following, we formulate a generic, unified model which encompasses prior approaches [Hen+17] (described earlier) and [Gao+17] (proposed concurrently with the MCN model). This allows us to explore and evaluate trade offs for different model components and extensions which then leads to higher performance. Unlike prior work, we consider a latent context variable which enables our model to better reason about temporal language.

Let the moment τ corresponding to the text query be the *base* moment and the set of other video moments T_τ be possible *context* moments for τ. We define a scoring function between the video moment and natural-language query by maximizing over all possible context moments $\tau' \in \mathrm{T}_\tau$,

$$s_\phi\left(v, q, \tau\right) = \max_{\tau' \in \mathrm{T}_\tau} f_\mathcal{S}\left(f_\mathcal{V}\left(v, \tau, \tau'\right), f_\mathcal{L}\left(q\right)\right), \tag{6.6}$$

where $f_\mathcal{V}$ and $f_\mathcal{L}$ are functions computing features over the video and language query, $f_\mathcal{S}$ is a similarity function, and ϕ are model parameters. This formulation is generic and trivially encompasses the MCN and TALL formulations by letting the set of possible context moments T_τ be their respective single-context moment. Figure 6.8 shows the generic structure of our model.

With this formulation, we seek to answer the following questions: (i) Which combination of model components performs best for the moment-retrieval task? Though our primary goal is localizing moments with temporal language, we believe a good base moment retrieval model is important for localizing moments with temporal language. (ii) How best to incorporate context for moment retrieval with temporal language? We first detail the different terms and outline different model design choices, where design choices marked with ***bold-italic font*** is ablated in Section 6.6. Components which are used in our final proposed Moment Localization with Latent Context (MLLC) model and prior models are summarized in Table **??**.

Video feature $f_\mathcal{V}$. The video feature $f_\mathcal{V} = \left(g\left(v, \tau\right), g\left(v, \tau'\right), f_\mathcal{T}\left(\tau, \tau'\right)\right)$ is a concatenation of visual features for the base $g\left(v, \tau\right)$ and context $g\left(v, \tau'\right)$ moments and endpoint features $f_\mathcal{T}\left(\tau, \tau'\right)$. To compute visual features g for a temporal region τ, per-frame features are averaged over the temporal region. Note that if the context moment consists of more than one contiguous temporal region,

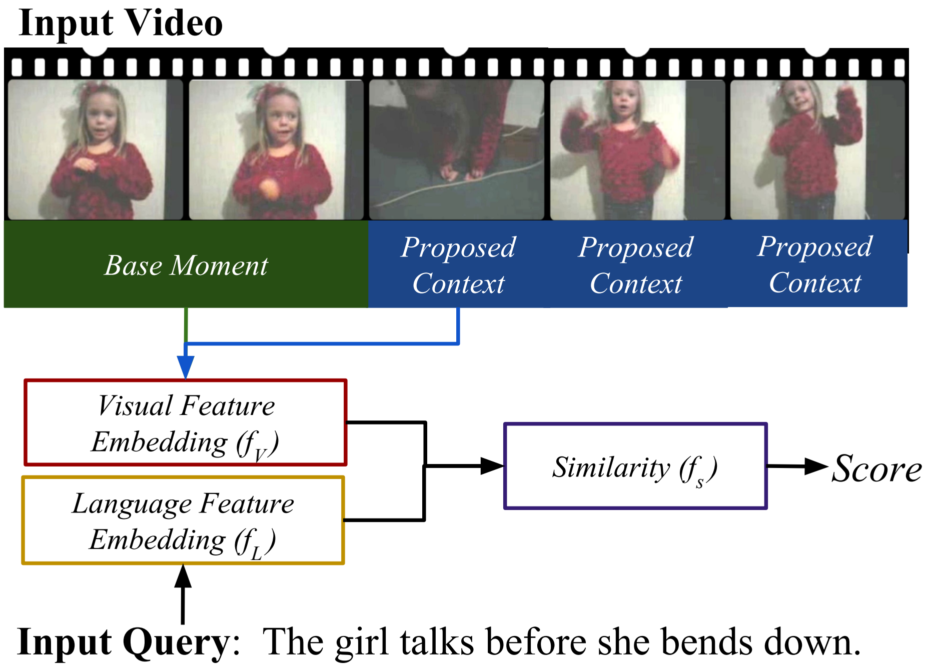

Input Video
Base Moment
Proposed Context
Proposed Context
Proposed Context
Visual Feature Embedding (f_V)
Language Feature Embedding (f_L)
Similarity (f_s)
Score
Input Query: The girl talks before she bends down.

Figure 6.8: Our model, Moment Localization with Latent Context (MLLC), takes a video and a text query as input and outputs the moment in the video corresponding to the query. MLLC considers many different *context* moments (blue) for a specific *base* moment (green).

	Endpoint Feature	Similarity	Context	Training Loss	Supervised Temp. Context
TALL [Gao+17]	None	TALL sim.	Before/After	TALL loss	None
MCN [Hen+17]	TEF	Distance-based	Global	Ranking	None
MLLC (ours)	**conTEF**	Normalized mult	**Latent**	Ranking	**Strongly sup.**

Table 6.5: Comparison of models. Bolded entries show our additions for localizing temporal language.

then the visual features are computed over each contiguous temporal region and then concatenated (c.f., before/after context in TALL, explained below). There are many choices for visual features. TALL [Gao+17] compares average fc_7 features (extracted from [SZ15]) to features extracted with C3D [Tra+15] and LSTM features [Don+15]. Surprisingly, C3D features only outperform average fc_7 features by a small margin. We use the visual features used in the MCN model [Hen+17], which are similar to the fc_7 features from [Gao+17], but included motion features as well, computed from optical flow (extracted with [Wan+16]). We then pass the extracted visual features through a MLP. Note that we learn separate embedding functions for RGB and optical flow inputs and combine scores from different input modalities using a late-fusion approach [Hen+17].

Endpoint feature $f_\mathcal{T}$. Modeling temporal context requires understanding how different temporal segments relate in time. [Hen+17] suggest including temporal endpoint features (***TEF***) $f_\mathcal{T} = \left(\tau^{(s)}, \tau^{(e)}\right)$ for the base moment which encode when the moment starts and ends to better localize sentences which include words like "first" and "last". Note that TALL [Gao+17] does not incorporate TEFs. In order to understand temporal relationships, it is important that models also include features which indicate when a context moment occurs. In addition to providing TEFs for base moments, we also experiment with concatenating TEFs for context moments (***conTEF***) $f_\mathcal{T} = \left(\tau^{(s)}, \tau^{(e)}, \tau'^{(s)}, \tau'^{(e)}\right)$.

Language feature $f_\mathcal{L}$. Text queries are transformed into a fixed-length vector with an LSTM [HS97]. Before inputting words into the LSTM, they are embedded in the Glove [PSM14] embedding space. The final layer of the LSTM is projected into the shared video-language embedding space with a fully connected layer. [Gao+17] considers LSTM language features and Skip-thought encoders. Our main goal is to study how context impacts moment localization with temporal language, so we use the LSTM features used on the original DiDeMo dataset.

Similarity $f_\mathcal{S}$. Given video $f_\mathcal{V}$ and language $f_\mathcal{L}$ features, we consider three ways to encode similarity between the features. Like [Hen+17], we consider a ***distance-based*** similarity $f_\mathcal{S} = (|f_\mathcal{V} - f_\mathcal{L}|^2)$. Second, we consider a fused-feature similarity (***mult***) where the Hadamard product $f_\mathcal{V} \odot f_\mathcal{L}$ between the two features are passed to a MLP. We also explore unit normalizing features before the Hadamard product (***normalized mult***). Finally, we consider the similarity (***TALL similarity***) which consists of the concatenation $(f_\mathcal{V}, f_\mathcal{L}, f_\mathcal{V} \odot f_\mathcal{L}, f_\mathcal{V} + f_\mathcal{L})$ and then passed to a MLP.

Context moments T_τ. We consider three sets of context moments. First, we consider the entire video as the context moment (***global***) following [Hen+17]. Second, we consider using the moments just before and after the base moment (***before/after***). Finally, we consider using the set of all possible moments (***latent*** context) which offers greatest flexibility in contextual reasoning.

Training loss. We consider two training losses. The first loss is the MCN ***ranking loss*** which encourages positive moment/query pairs to have a smaller distance in a shared embedding space than negative moment/query pairs. To sample negative moment/sentence pairs, they consider negative moments *within* a specific video (called intra-video negative moments) and negative moments in different videos (called inter-video negative moments). This sampling strategy leads to a small improvement in performance (approximately one point on all metrics) when compared to just using intra-video negative moments. We also consider the alignment loss used in TALL (***TALL loss***) which is the sum of two log-logistic functions over positive and negative training query/moment pairs (intra-video negatives are used).

Supervising context moments. For the temporal sentences in our newly collected dataset (Section 6.4), we have access to the ground-truth context moment during training. Thus, we can contrast a ***weakly supervised*** setting in which we optimize over the unknown latent context moments during learning and inference to a ***strongly supervised setting***.

Implementation details. Candidate base and context moments coincide to the pre-segmented five-second segments used when annotating DiDeMo. Moments may consist of any contiguous set of five-second segments. For a 30-second video partitioned into six five-second segments, there are 21 possible moments. All models were implemented in Caffe [Jia+14] and optimized with SGD. Models were trained for $\sim$ 90 epochs with an initial learning rate of 0.05, which decreases every 30 epochs. Code is publicly released[7].

6.6 Results: Moment Localization with Latent Context

Evaluation Method. We follow the evaluation protocol defined for the DiDeMo dataset [Hen+17] over all possible combinations of the five-second video segments. We report rank at one (R@1), rank at five (R@5), and mean intersection over union (mIOU) using their aggregator over three out of the four human annotators. We compare our models on TEMPO-TL, TEMPO-HL, and the DiDeMo dataset. When training our models, we combine the DiDeMo dataset with TEMPO-TL or TEMPO-HL. This enables our model to concurrently learn to localize the simpler DiDeMo sentences with more complex TEMPO sentences.

Baselines. We compare to the two recently proposed approaches for video moment localization: MCN [Hen+17] and TALL [Gao+17]. We adapt the implementation of TALL [Gao+17] to the DiDeMo dataset in three ways. First, we do not include the temporal localization loss required to regress to specific start and end points as DiDeMo, and thus also TEMPO, is pre-segmented, so the model does not need to compute exact start and end points. Second, the original TALL model uses C3D features. For a fair comparison we train both models with the same RGB and flow features extracted as was done for the original MCN model. Finally, the MCN model proposes temporal endpoint features (TEF) to indicate when a proposed moment occurs within a video. We train TALL with and without the TEF and show that TEF improves performance on the original DiDeMo dataset.

Ablations. To ablate our proposed latent context, we compare to other models which share the same MLLC base network. We consider the MLLC model with global context and before/after context. We also train a model with weakly supervised (WS) latent context and strongly supervised (SS) latent context. We also train models both with and without context TEF (conTEF).

The MLLC Base Model. We first ablate our MLLC base model (Table 6.6). We train our models on TEMPO-TL and DiDeMo and evaluate on the original DiDeMo dataset. All models are trained with global context. We find that the ranking loss is preferable on the DiDeMo dataset

[7] https://people.eecs.berkeley.edu/~lisa_anne/tempo.html

(compare lines 1 and 2) and that TALL-similarity performs better than the distance based similarity of the MCN model (compare lines 1 and 5). A simpler version of the TALL-similarity, in which the concatenated element wise multiplication, element wise sum, and concatenation is replaced by a single normalized elementwise multiplication, increases R@1 by almost one point and increases mIoU by over two points (compare lines 5-7). We call our best model the MLLC-Base model (line 7). Our MLLC-Base model performs better than previous models (MCN line 1 and TALL line 3).

	Model	Similarity	Training Loss	R@1	R@5	mIoU
1	MCN	Dist.-based	Ranking	26.63	73.38	41.14
2	MCN	Dist.-based	TALL	23.89	76.54	35.69
3	TALL	TALL-sim.	TALL	8.04	36.32	22.68
4	TALL w/TEF	TALL-sim.	TALL	23.56	72.74	35.58
5	MCN	TALL-sim	Ranking	27.52	**79.07**	41.48
6	MCN	Mult	Ranking	28.19	78.97	43.21
7	MLLC-Base	Norm. Mult	Ranking	**28.37**	78.64	**43.65**

Table 6.6: To select our base network, we consider different variants on the two previously proposed moment retrieval methods, TALL [Gao+17] and MCN [Hen+17]. Results reported on val.

Results: TEMPO - TL. We first compare different moment localization models on TEMPO - TL (Table 6.7). In particular, our model performs well on "before" and "after" words. Additionally, our MLLC model with global context outperforms both the MCN model [Hen+17] and the TALL [Gao+17] model when considering all sentence types, verifying the strength of our base MLLC model.

Comparing MLLC with global context and MLLC with before/after context (compare row 4 and 5), we note that before/after context is important for localizing "before" and "after" moments. However, our model with strong supervision (row 9) outperforms the model trained with before and after context, suggesting that learning to reason about which context moment is correct (as opposed to being explicitly provided with the context before and after the moment) is beneficial. We note that strong supervision (SS) outperforms weak supervision (WS) (compare rows 7 and 9) and that the context TEF is important for best performance (compare rows 8 and 9).

We note that though the MLLC-global model outperforms our full model for "then" on TEMPO-TL, our full model performs better on then for the TEMPO-HL (Table 6.9). One possibility is that the "then" moments in TEMPO-TL do not require context to properly localize the moment. Because TEMPO-TL is constructed from DiDeMo sentences, constituent sentence parts are *referring*. For example, given an example sentence from TEMPO-TL (e.g., "The cross is seen for the first time *then* window is first seen in room"), the model does not need to reason about the ordering of "cross seen for the first time" and "window is seen for the first time" because both moments only happen once in the video. In contrast, when considering the sentence "The adult hands the little boy a stick *then* they begin to walk" (from Figure 6.7), "begin to walk" could refer to multiple

		TEMPO - Template Language (TL)										
		DiDeMo		Before		After		Then		**Average**		
		R@1	mIoU	R@1	mIoU	R@1	mIoU	R@1	mIoU	R@1	R@5	mIoU
1	Frequency Prior	10.71	20.67	17.85	24.22	22.42	25.76	0.00	24.73	12.74	52.58	23.84
2	MCN	24.85	37.92	32.28	38.67	26.08	35.44	25.07	53.94	27.07	73.36	41.49
3	TALL	20.95	32.09	27.13	32.41	26.30	34.27	4.84	36.75	19.80	64.66	33.88
4	MLLC- Global	26.32	40.37	31.92	38.26	25.37	35.59	**27.53**	**57.08**	27.78	74.14	42.82
5	MLLC B/A	26.04	39.60	34.04	40.46	28.50	38.18	25.60	54.37	28.54	74.92	43.15
6	MLLC (WS)	26.57	40.99	30.56	37.64	24.76	35.10	26.95	56.49	26.95	74.18	42.55
7	MLLC (WS + conTEF)	25.87	40.37	32.01	39.51	24.31	33.94	24.98	55.22	26.79	74.04	42.27
8	MLLC (SS)	26.09	40.12	28.45	34.38	23.79	33.92	24.27	55.00	25.65	73.60	40.86
9	MLLC (SS + conTEF)	**27.46**	**41.20**	**35.31**	**41.81**	**29.38**	**38.90**	26.83	54.97	**29.74**	**76.76**	**44.22**

Table 6.7: Comparison of different model performance for different temporal words on TEMPO - TL on our test set. We report scores for the three temporal words in TEMPO - TL as well as on the original DiDeMo dataset. We find that our model performs best when considering all sentence types. B/A indicated before/after context, WS indicates weak context supervision, and SS indicates strong context supervision.

	Before		After		Then	
Context	R@1	mIoU	R@1	mIoU	R@1	mIoU
Global	-1.07	-2.72	-7.59	-6.75	43.30	31.57
Before/After	2.77	2.03	11.47	12.08	42.92	29.09
Latent	7.78	37.55	8.58	10.39	50.09	33.64

Table 6.8: Difference between performance on full dataset and set on which reference moments are localized properly for different methods on TEMPO-TL.

video moments. Consequently, our model must reason about the temporal ordering of reference moments to properly localize the video moment.

On TEMPO - TL, sentences differ from original DiDeMo sentences solely because of the use of temporal words. Thus, we can do a controlled study of how well models understand temporal words. If a model has good temporal reasoning, then if it can localize a reference moment “the dog jumps” it should be easier for the model to localize the moment “the dog sits after the dog jumps”. To test whether models are capable of this, we look at only sentences in TEMPO - TL where the model has correctly localized the corresponding context moment in DiDeMo (Table 6.8). We report the *difference* in performance when considering only sentences in which temporal context was properly localized and all sentences. On our model, performance on all three temporal word types increases when the context moment can be properly localized. When considering global context, performance on “before” and “after” actually decreases, suggesting global context does not understand temporal reasoning well. Finally, even when the context is correctly localized, there is still ample room for improvement on all three sentence types motivating future work on temporal

TEMPO - Human Language (HL)

	DiDeMo		Before		After		Then		While		**Average**		
	R@1	mIoU	R@1	mIoU	R@1	mIoU	R@1	mIoU	R@1	mIoU	R@1	R@5	mIoU
Frequeny Prior	19.43	25.44	29.31	51.92	0.00	0.00	0.00	7.84	4.74	12.27	10.69	37.56	19.50
MCN	26.07	39.92	26.79	51.40	**14.93**	34.28	18.55	47.92	10.70	35.47	19.4	70.88	41.80
TALL + TEF	21.79	33.55	25.91	49.26	14.43	32.62	2.52	31.13	8.1	28.14	14.55	60.69	34.94
MLLC - Global	27.01	41.72	27.42	52.22	14.10	34.33	18.40	49.17	10.86	35.36	19.56	71.23	42.56
MLLC - B/A	26.47	40.39	31.95	55.89	**14.93**	34.78	17.36	47.52	**11.32**	35.52	20.40	70.97	42.82
MLLC (Ours)	**27.38**	**42.45**	**32.33**	**56.91**	14.43	**37.33**	**19.58**	**50.39**	10.39	**35.95**	**20.82**	**71.68**	**44.57**
MLLC Context Sup. Test	27.39	42.25	52.58	80.37	36.48	75.79	36.05	70.51	10.39	35.87	32.58	79.86	60.96

Table 6.9: Comparison of different model performance on TEMPO - HL on the test set. "MLLC - Global" indicates our model with global context and "MLLC - B/A" indicated MLLC with before/after context.

reasoning for moment retrieval.

Results: TEMPO - HL. Figure 6.9 shows an example of how MLLC can not only localize the original moment described in DiDeMo ("a cate jumps up and spazzes out"), but also moments which are related with various temporal relationships. Table 6.9 compares performance on TEMPO - HL. We compare our best-performing model from training on the TEMPO-TL (strongly supervised MLLC and conTEF) to prior work (MCN and TALL) and to MLLC with global and before/after context. Performance on TEMPO-HL is considerably lower than TEMPO-TL suggesting that TEMPO-HL is harder than TEMPO-TL.

On TEMPO - HL, we observe similar trends as on TEMPO-TL. When considering all sentence types, MLLC has the best performance across all metrics. In particular, our model has the strongest performance for all sentence types considering the mIoU metric. In addition to performing better on temporal words, our model also performs better on the original DiDeMo dataset. As was seen in TEMPO-TL, including before/after context performs better than our model trained with global context for both "before" and "after" words.

The final row of Table 6.9 shows an upper bound in which the ground truth context is used at test time instead of the latent context. We note that results improve for "before", "after", and "then", suggesting that learning to better localize context will improve results for these sentence types.

Localizing Context Fragments. TEMPO-HL sentences can be broken into two parts: a base-sentence fragment (which refers to the base moment), and a context-sentence fragment (which refers to the context moment). For example, for the sentence "The girl holds the ball before throwing it,", "the girl holds the ball" is the base fragment and "throwing it" is the context fragment. A majority of the "before" and "after" sentences in TEMPO-HL are of the form "X before (or after) Y", so we can determine a list of sentence fragments by splitting sentences based on the temporal word. Given "before" and "after" sentences, we determine the ground truth context fragment

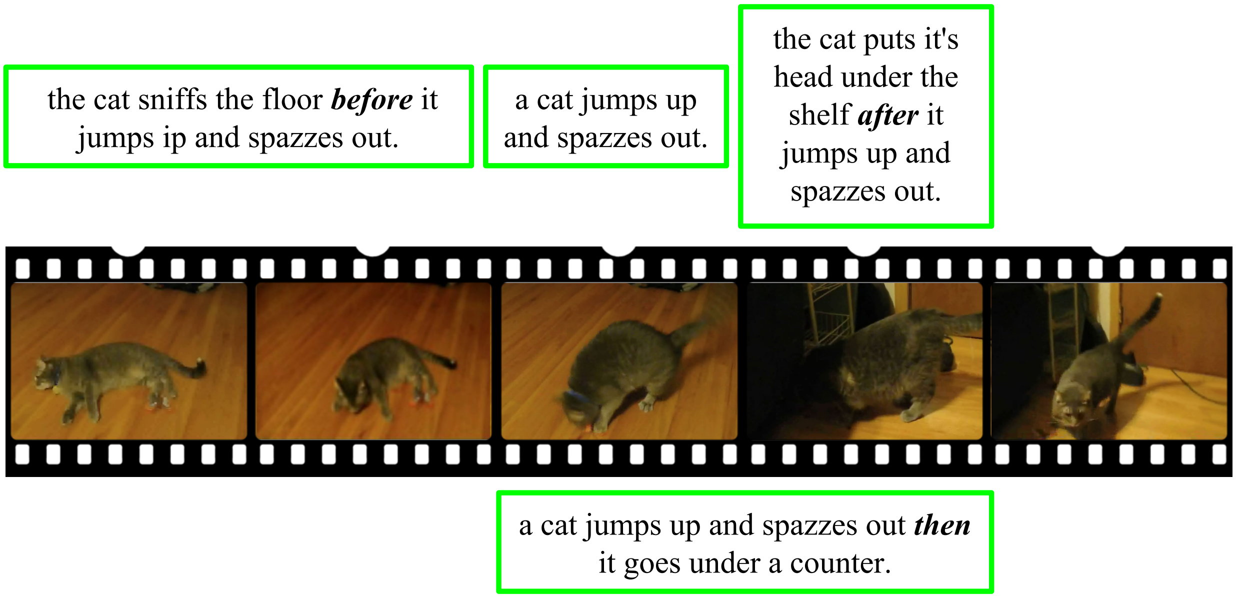

the cat sniffs the floor *before* it jumps ip and spazzes out.
a cat jumps up and spazzes out.
the cat puts it's head under the shelf *after* it jumps up and spazzes out.
a cat jumps up and spazzes out *then* it goes under a counter.

Figure 6.9: Moment localization predictions on TEMPO - HL using our model. MLLC can localize the original DiDeMo moment ("a cat jumps up and spazzes out") as well as other moments with a temporal relation to the original DiDeMo moment.

	Before		After	
	R@1	mIoU	R@1	mIoU
Context Fragment	25.16	32.94	23.05	27.64
Full Sentence	**27.55**	**35.70**	**32.67**	**40.39**

Table 6.10: Comparison of different methods to localize context fragments (e.g., the text "she bends down" in the sentence "the girl talks after she bends down"). We compare localizing fragments with the MLLC model to localizing fragments with the latent context considered when localizing the whole query.

by considering which reference moment was given to annotators. We can then measure how well models localize context fragments. Table 6.10 compares two approaches to localizing context fragments: inputting just the context fragment into MLLC and reporting the context used by MLLC when inputting the entire query into our model. We find that our model reliably selects the correct context fragments, most likely because it can properly exploit temporal understanding of how the context fragment relates to the base fragment.

Visualizing Context. In addition to a localized query, we can also visualize which context moment the temporal query refers to. Figure 6.10 shows predicted moments and their corresponding context moments. For the query "The girl with a hat takes a drink before the girl without a hat waves", the little girl in the hat drinks twice, but our model correctly localizes the time she drinks

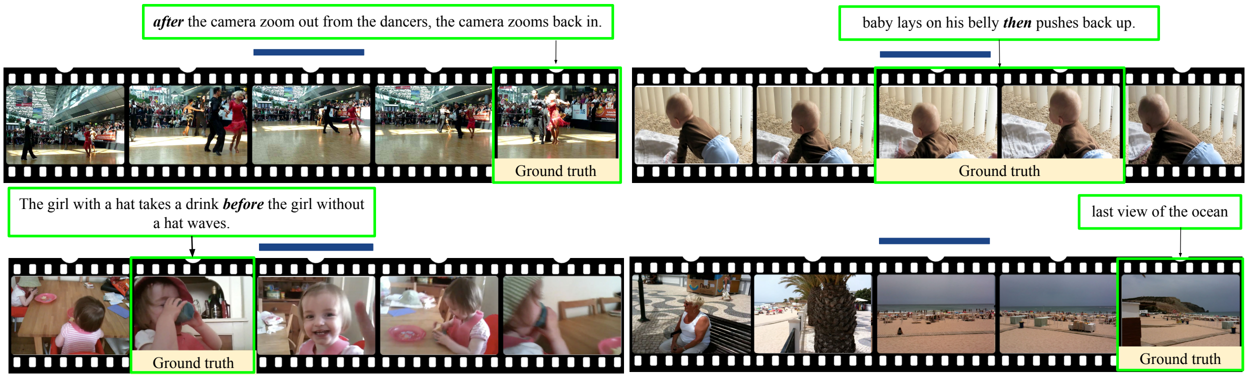

after the camera zoom out from the dancers, the camera zooms back in.
Ground truth
baby lays on his belly *then* pushes back up.
Ground truth
The girl with a hat takes a drink *before* the girl without a hat waves.
Ground truth
last view of the ocean
Ground truth

Figure 6.10: Moment localization predictions on TEMPO - HL using our model. In addition to the localized query, we show the selected context segment (blue line) that our model considers when localizing the query.

before the other girl waves. For the moment "after the camera zooms out from the dancers, the camera zooms back in", the camera zooms out well before zooming back in, but our model is still able to correclty localize the proper context. For the moment "baby lays on his belly then pushes up", the model localizes "baby lays on his belly" as the context. Finally, for "last view of the ocean", no temporal word is used, but the model localizes a reasonable context moment: the first time the ocean is viewed in the video.

We show promising results on both TEMPO-TL and TEMPO-HL, but there is potential improvement for building better frameworks for understanding temporal language. In Table 6.9, strongly supervising context at test time improves overall results, suggesting that models which can better localize context text will outperform our current model. Additionally, in Table 6.8, even when the MLLC model can properly localize context, it does not always properly localize temporal sentences indicating that improved temporal reasoning can also improve our results. We believe our dataset, analysis, and method are an important step towards better moment retrieval models that effectively reason about temporal language.

6.7 Discussion

6.7.1 Related Work

Localizing moments in video with natural language is related to other vision tasks including video retrieval, video summarization, video description and question answering, and natural language object retrieval. Though large scale datasets have been collected for each of these tasks, before the collection of DiDeMo and TEMPO, none fit the specific requirements needed to learn how to localize moments in video with natural language. Our work on understanding temporal relationships in video is related to work in the natural lnaguage processing community on understanding temporal relationships in text as well as work on understanding spatial relationships in images.

Video Retrieval with Natural Language. Natural language video retrieval methods aim to retrieve a specific video given a natural language query. Current methods [Ota+16; TTS16; Xu+15b] incorporate deep video-language embeddings similar to image-language embeddings proposed by [Fro+13; Soc+14]. Our method also relies on a joint video-language embedding. However, to identify when events occur in a video, our video representation integrates local and global video features as well as temporal endpoint features which indicate when a candidate moment occurs within a video.

Some work has studied retrieving temporal segments within a video in constrained settings. For example, [TR09] considers retrieving video clips from a home surveillance camera using text queries which include a fixed set of spatial prepositions ("across" and "through") whereas [Lin+14a] considers retrieving temporal segments in 21 videos from a dashboard car camera. In a similar vein, [Ala+16; Boj+15; Sen+15] consider aligning textual instructions to videos. However, methods aligning instructions to videos are restricted to structured videos as they constrain alignment by instruction ordering. In contrast, in DiDeMo we consider localizing moments in an unconstrained open-world dataset with a wide array of visual concepts.

Video Summarization. Video summarization algorithms isolate temporal segments in a video which include important/interesting content. Though most summarization algorithms do not include textual input ([BI07; GGVG15; GSC16; Yan+15; YMR16]), some use text in the form of video titles [Liu+15; Son+15] or user queries in the form of category labels to guide content selection [SGS16]. [YFFF14] collects textual descriptions for temporal video chunks as a means to evaluate summarization algorithms. However, these datasets do not include referring expressions and are limited in scope which makes them unsuitable for learning moment retrieval in an open-world setting.

Video Description and Question Answering (QA). Video description models learn to generate textual descriptions of videos given video-description pairs. Contemporary models integrate deep video representations with recurrent language models [Pan+16; RRS15; Ven+15a; Ven+15b; Yu+15b]. Additionally, [Tap+16] proposed a video QA dataset which includes question/answer pairs aligned to video shots, plot synopsis, and subtitles.

YouTube and movies are popular sources for joint video-language datasets. Video description datasets collected from YouTube include descriptions for short clips of longer YouTube videos [CD11; Xu+16]. Other video description datasets include descriptions of short clips sourced from full length movies [Roh+17a]. However, though YouTube clips and movie shots are sourced from longer videos, they are not appropriate for localizing distinct moments in video for two reasons. First, descriptions about selected shots and clips are not guaranteed to be referring. For example, a short YouTube video clip might include a person talking and the description like "A woman is talking". However, the entire video could consist of a woman talking and thus the description does not uniquely refer to the clip. Second, many YouTube videos and movies are edited, which means "boring" content which may be important to understand for applications like retrieving video segments from personal videos might not be present.

Natural Language Object Retrieval. Natural language object retrieval [Hu+16; Mao+16] can be seen as an analogous task to ours, where natural language phrases are localized spatially in images, rather than temporally in videos. Despite similarities to natural language object retrieval, localizing video moments presents unique challenges. For example, it often requires comprehension of temporal indicators such as “first” as well as a better understanding of activities. Datasets for natural language object retrieval include *referring* expressions which can uniquely localize a specific location in a image. Descriptions in DiDeMo uniquely localize distinct moments and are thus also referring expressions.

Language Grounding in Images and Videos. [Plu+15; Roh+16; Soc+14] tackle the task of object grounding in which sentence fragments in a description are localized to specific image regions. Work on language grounding in video is much more limited. Language grounding in video has focused on spatially grounding objects and actions in a video [Lin+14a; YS13], or aligning textual phrases to temporal video segments [Reg+13; TR09]. However prior methods in both these areas ([TR09; YS13]) severely constrain natural language vocabulary (e.g., [YS13] only considers four objects and four verbs) and consider constrained visual domains in small datasets (e.g., 127 videos from a fixed laboratory kitchen [Reg+13] and [Lin+14a] only includes 520 sentences). In contrast, DiDeMo and TEMPO offer a unique opportunity to study temporal language grounding in an open-world setting with a diverse set of objects, activities, and attributes.

Temporal Language. Prior work on temporal language processing has considered building explicit logical frameworks to process temporal prepositions like “during” or “until” ([PH04], [Kon08]). We do not derive a particular temporal logic, but rather learn to understand temporal language in a data driven fashion. Furthermore, we specifically consider how to understand temporal words commonly used when referring to video content. Other work has modeled dynamics for words which represent a change of state (e.g., “pick up”) ([Sis01], [Yu+15a]) in limited environments. Though we limit the selection of temporal words in our study, the natural language in our data is open-world describing diverse events and how they relate to each other in video. Interpretation of temporal expressions in text (“The game happened on the 19^{th}”) is a widely studied task ([AMJ12], [ZSC17]). Our work is distinctly different from this line of work as we specifically study temporal prepositions and how they refer to video.

Modeling Visual Relationships. A variety of papers have considered modeling spatial relationships in natural images [DZL17; Hu+17b; Pey+17; Plu+17]. Our approach is analogous to this in the temporal domain; we hope to localize moments in videos. CLEVR, a synthetic visual question answering (VQA) dataset [Joh+17a], was created to allow researchers to systematically study the ability of models to perform complex reasoning. Our dataset is partially motivated by the success of CLEVR to enable researchers to study reasoning abilities of different models in a controlled setting. In contrast to CLEVR we consider a more diverse visual input in the form of real videos.

In the video domain, the TGIF-QA [Jan+17] and Mario-QA [Mun+16] datasets provide opportunities to study temporal reasoning for the task of VQA. The TGIF-QA dataset considers three types of temporal questions: before/after questions, repetition count, and determining a repeating action. Each question is accompanied by multiple choice answers. Videos we consider are much longer (25-30s as opposed to an average of 3.1s) which makes the use of temporal reasoning much more important. The MarioQA dataset is an additional VQA dataset designed to gauge temporal reasoning of VQA systems. Both TGIF-QA and MarioQA datasets include template-based natural language queries. In this paper, we consider synthetic queries similar to TGIF-QA and MarioQA, but also include human language queries. In addition, unlike the MarioQA dataset, that consists of synthetic data constructed from gameplay videos, our dataset consists of real visual inputs, and includes temporal grounding of natural language phrases. Finally, neither TGIF-QA nor MarioQA include temporal localization.

6.7.2 Concurrent and Future Work

Since the original publication of our DiDeMo dataset and MCN model, others have considered our proposed problem statement. In particular, concurrently with our work, [Gao+17] proposed the task of natural language moment localization as well as a new dataset, Charades-STA. Charades-STA is built off the Charades [Sig+16] dataset, which consists of short descriptions of videos (not temporally grounded in the video) as well as temporal annotations for 157 different action classes. To build off the Charades dataset, [Gao+17] linked sentences to activity localization annotations (e.g., if a video is annotated with the sentence “A man walks to the door” and the activity “walk” is annotated, authors connected the sentence to the time stamps of the activity walk). Though this is advantageous because sentences are annotated with more fine-grained start and end points, activities necessarily align with one of the 157 action classes in Charades. Both the release of our DiDeMo dataset and the Charades-STA dataset have encouraged more research on the topic of moment loclalization with natural language.

Recently, a variety of work has improved performance for moment localization on the DiDeMo dataset [Che+19; Zha+18; Liu+18a; Liu+18b; Che+18b]. Interesting contributions include modeling temporal relations between moments as structured graphs [Zha+18] or extending modular networks ([And+16c; Joh+17b; Hu+17a]) to temporal relationships [Liu+18a]. In our initial work on the DiDeMo and TEMPO datasets, we found that it was important to provide the model with the start and end points of different candidate moments. Though this improved results, conceptually it is less satisfying as it indicates that the model might rely more on *when* moments occurred in a video than semantic content. However, more recent work [Che+18b] is capable of outperforming our MCN model without access to temporal endpoint features. In addition to the moment localization task as we defined in our original paper, others have used the DiDeMo dataset for related tasks. For example [ZHS18] considers paragraph to video retrieval (and vice versa) with the DiDeMo dataset.

Moment localization with natural language is a recent task, and there is still considerable work to be done. In particular, understanding videos requires understanding objects, actions, and attributes *as well as* complex temporal and spatial relationships between different visual entities. It is unlikely that a model can learn about all objects and actions from the 10,000 videos in DiDeMo. Though most models use networks that have been pre-trained for related tasks such as object and activity recognition, it seems that systems that can integrate knowledge from a variety of other tasks in a more sophisticated fashion should perform better on the moment localization task. Additionally, work on moment localization does not generally model spatial relationships in video which could be important for better performance. In order to make the moment localization task more realistic, longer videos should also be considered. Finally, so far in this thesis we have considered models which can output language given a visual input, and models that can retrieve visual information given a language input, but for AI systems to truly *interact* with human users, they should be capable of both these tasks. We will discuss this as potential future work in our final discussion on visual understanding through natural language.

Chapter 7

Conclusion

In this thesis we first considered how to build more robust image captioning systems. In Chapter 2, we formulated the task of novel object captioning and introduced the first deep caption systems with the capability of learning to describe novel objects from classification data. We then considered bias in image captioning. In Chapter 3 we analyzed and measured hallucination in image captioning and in Chapter 4 we proposed the Equalizer model which produces less biased descriptions in regard to gender. In Chapter 5, we introduced a paradigm for textual explanations of visual decisions. We proposed a novel training procedure to generate discriminative text and the phrase critic model to ground explanatory visual evidence in an image. Finally, in Chapter 6, we considered a new task at the intersection of language and vision: moment localization with natural language. We proposed new datasets for this task as well as models to localize video moments with natural language.

Beyond Image Captioning and Moment Localization. This thesis focuses on image captioning and moment localization tasks. Though there are immediate applications of these tasks, such as visual description for the blind or retrieval of videos in databases, for natural language to truly benefit AI and human users, machines must be capable of engaging in dialog with humans and acting in the real world. For example, consider telling an AI agent to perform some task, like fetching a red cup from the kitchen. The AI agent might need to engage in dialog with the human clarifying the command (AI: “The red cup looks a little dirty, should I still bring it over?” Human: “No, grab the blue one.” AI: “Where is it?”) and, of course, act in the world by navigating to the kitchen, finding the correct cup, and bringing it to the human. In other words, the AI agent needs to engage in *dialog* and should be *embodied*.

Visual dialog [Das+17b; Kot+19] and embodiment [And+18b; Gor+18; Das+18] are emerging areas of research and are gaining interest in the vision and language community. For these more complex tasks, ideas of compositionality, understanding and mitigating bias, and explainability are perhaps even more important than in image captioning. One advantage of the DCC and NOC models described in Chapter 2 is that they successfully perform a more complex task (image captioning) by learning from lower level tasks (object recognition and unsupervised language modeling). Being able to effectively transfer knowledge from lower level tasks will likely be even more important as the research community tackles more complex tasks and training data becomes more costly and difficult to collect. As the input space becomes more complex (e.g., in dialog a human and AI user may build up a history of interaction), ensuring our models are right for the right reason, and do not just perform well on some benchmark metric, will be increasingly important. Though we may not be able to directly test how an embodied AI agent will behave in every scenario it may encounter, if the agent can make decisions for the right reasons, we may trust it will make better decisions in new scenarios. Finally, to both debug more complex systems and for AI agents to effectively cooperate with humans, explanations are essential. Thus, as the vision and language community continues to pursue more challenging and impactful tasks, the ideas explored in this thesis should be expanded to enable more robust and reliable systems.

Bibliography

[AA19] Stephan Alaniz and Zeynep Akata. “XOC: Explainable Observer-Classifier for Explainable Binary Decisions”. In: *arXiv preprint arXiv:1902.01780* (2019).

[Ade+18] Julius Adebayo, Justin Gilmer, Michael Muelly, Ian Goodfellow, Moritz Hardt, and Been Kim. “Sanity checks for saliency maps”. In: *Advances in Neural Information Processing Systems*. 2018, pp. 9505–9515.

[AGJ18] Peter Anderson, Stephen Gould, and Mark Johnson. “Partially-Supervised Image Captioning”. In: *Advances in Neural Information Processing Systems*. 2018, pp. 1879–1890.

[Agr+17] Aishwarya Agrawal, Aniruddha Kembhavi, Dhruv Batra, and Devi Parikh. “C-vqa: A compositional split of the visual question answering (vqa) v1. 0 dataset”. In: *arXiv preprint arXiv:1704.08243* (2017).

[Agr+18] Harsh Agrawal, Karan Desai, Xinlei Chen, Rishabh Jain, Dhruv Batra, Devi Parikh, Stefan Lee, and Peter Anderson. “nocaps: novel object captioning at scale”. In: *arXiv preprint arXiv:1812.08658* (2018).

[AK16] Jacob Andreas and Dan Klein. “Reasoning about pragmatics with neural listeners and speakers”. In: *arXiv preprint arXiv:1604.00562* (2016).

[Ala+16] Jean-Baptiste Alayrac, Piotr Bojanowski, Nishant Agrawal, Josef Sivic, Ivan Laptev, and Simon Lacoste-Julien. “Unsupervised learning from narrated instruction videos”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016.

[AMJ12] Gabor Angeli, Christopher D Manning, and Daniel Jurafsky. “Parsing time: Learning to interpret time expressions”. In: *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics. 2012, pp. 446–455.

[And+16a] Peter Anderson, Basura Fernando, Mark Johnson, and Stephen Gould. “Spice: Semantic propositional image caption evaluation”. In: *European Conference on Computer Vision*. Springer. 2016, pp. 382–398.

[And+16b] Jacob Andreas, Marcus Rohrbach, Trevor Darrell, and Dan Klein. "Learning to Compose Neural Networks for Question Answering". In: *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*. 2016.

[And+16c] Jacob Andreas, Marcus Rohrbach, Trevor Darrell, and Dan Klein. "Neural module networks". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2016, pp. 39–48.

[And+17] Peter Anderson, Basura Fernando, Mark Johnson, and Stephen Gould. "Guided open vocabulary image captioning with constrained beam search". In: *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 2017.

[And+18a] Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. "Bottom-up and top-down attention for image captioning and visual question answering". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018, pp. 6077–6086.

[And+18b] Peter Anderson, Qi Wu, Damien Teney, Jake Bruce, Mark Johnson, Niko Sünderhauf, Ian Reid, Stephen Gould, and Anton van den Hengel. "Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018, pp. 3674–3683.

[Arg+07] Shlomo Argamon, Moshe Koppel, James W Pennebaker, and Jonathan Schler. "Mining the blogosphere: Age, gender and the varieties of self-expression". In: *First Monday* 12.9 (2007).

[Bau+17] David Bau, Bolei Zhou, Aditya Khosla, Aude Oliva, and Antonio Torralba. "Network dissection: Quantifying interpretability of deep visual representations". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2017, pp. 6541–6549.

[BB13] Thomas Berg and Peter Belhumeur. "How Do You Tell a Blackbird from a Crow?" In: *ICCV*. 2013, pp. 9–16.

[BI07] Oren Boiman and Michal Irani. "Detecting Irregularities in Images and in Video". In: *International Journal of Computer Vision (IJCV)* (2007).

[BKL09] Steven Bird, Ewan Klein, and Edward Loper. *Natural language processing with Python*. "O'Reilly Media, Inc.", 2009.

[BL05] Satanjeev Banerjee and Alon Lavie. "METEOR: An automatic metric for MT evaluation with improved correlation with human judgments". In: *Proceedings of the ACL workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*. Vol. 29. 2005, pp. 65–72.

[Blu+18] Phil Blunsom, Oana-Maria Camburu, Thomas Lukasiewicz, and Tim Rocktäschel. "e- SNLI: Natural Language Inference with Natural Language Explanations". In: (2018).

[BM14] Or Biran and Kathleen McKeown. “Justification narratives for individual classifications”. In: *Proceedings of the AutoML workshop at ICML 2014*. 2014.

[Boj+15] Piotr Bojanowski, Rémi Lajugie, Edouard Grave, Francis Bach, Ivan Laptev, Jean Ponce, and Cordelia Schmid. “Weakly-supervised alignment of video with text”. In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. 2015.

[Bol+16] Tolga Bolukbasi, Kai-Wei Chang, James Y Zou, Venkatesh Saligrama, and Adam T Kalai. “Man is to computer programmer as woman is to homemaker? debiasing word embeddings”. In: *Advances in Neural Information Processing Systems (NIPS)*. 2016, pp. 4349–4357.

[BS16] Solon Barocas and Andrew D Selbst. “Big data’s disparate impact”. In: *California Law Review* 104 (2016), p. 671.

[Buo17] Joy Adowaa Buolamwini. “Gender shades: intersectional phenotypic and demographic evaluation of face datasets and gender classifiers”. PhD thesis. Massachusetts Institute of Technology, 2017.

[Bur+11] John D Burger, John Henderson, George Kim, and Guido Zarrella. “Discriminating gender on Twitter”. In: *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics. 2011, pp. 1301–1309.

[Can87] John Canny. “A computational approach to edge detection”. In: *Readings in computer vision*. Elsevier, 1987, pp. 184–203.

[Car+12] Joao Carreira, Rui Caseiro, Jorge Batista, and Cristian Sminchisescu. “Semantic segmentation with second-order pooling”. In: *European Conference on Computer Vision*. Springer. 2012, pp. 430–443.

[Car+15] Rich Caruana, Yin Lou, Johannes Gehrke, Paul Koch, Marc Sturm, and Noemie Elhadad. “Intelligible models for healthcare: Predicting pneumonia risk and hospital 30-day readmission”. In: *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM. 2015, pp. 1721–1730.

[CD11] David L Chen and William B Dolan. “Collecting highly parallel data for paraphrase evaluation”. In: *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*. 2011.

[Cha+18] Arjun Chandrasekaran, Viraj Prabhu, Deshraj Yadav, Prithvijit Chattopadhyay, and Devi Parikh. “Do explanations make VQA models more predictable to a human?” In: *arXiv preprint arXiv:1810.12366* (2018).

[Che+15] Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and C Lawrence Zitnick. “Microsoft COCO captions: Data collection and evaluation server”. In: *arXiv preprint arXiv:1504.00325* (2015).

[Che+18a] Chaofan Chen, Oscar Li, Alina Barnett, Jonathan Su, and Cynthia Rudin. “This looks like that: deep learning for interpretable image recognition”. In: *arXiv preprint arXiv:1806.10574* (2018).

[Che+18b] Jingyuan Chen, Xinpeng Chen, Lin Ma, Zequn Jie, and Tat-Seng Chua. “Temporally grounding natural sentence in video”. In: *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. 2018, pp. 162–171.

[Che+19] Jingyuan Chen, Lin Ma, Xinpeng Chen, Zequn Jie, and Jiebo Luo. “Localizing Natural Language in Videos”. In: (2019).

[Chi+94] Michelene TH Chi, Nicholas De Leeuw, Mei-Hung Chiu, and Christian LaVancher. “Eliciting self-explanations improves understanding”. In: *Cognitive science* 18.3 (1994), pp. 439–477.

[Cor+06] Mark G Core, H Chad Lane, Michael Van Lent, Dave Gomboc, Steve Solomon, and Milton Rosenberg. “Building explainable artificial intelligence systems”. In: *Proceedings of the national conference on artificial intelligence*. Vol. 21. 2. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999. 2006, p. 1766.

[Dai+17] Bo Dai, Sanja Fidler, Raquel Urtasun, and Dahua Lin. “Towards diverse and natural image descriptions via a conditional gan”. In: *Proceedings of the IEEE International Conference on Computer Vision*. 2017, pp. 2970–2979.

[Das+13] Pradipto Das, Chenliang Xu, Richard F Doell, and Jason J Corso. “A thousand frames in just a few words: Lingual description of videos through latent topics and sparse object stitching”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2013.

[Das+17a] Abhishek Das, Harsh Agrawal, Larry Zitnick, Devi Parikh, and Dhruv Batra. “Human attention in visual question answering: Do humans and deep networks look at the same regions?” In: *Computer Vision and Image Understanding* 163 (2017), pp. 90–100.

[Das+17b] Abhishek Das, Satwik Kottur, Khushi Gupta, Avi Singh, Deshraj Yadav, José MF Moura, Devi Parikh, and Dhruv Batra. “Visual dialog”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2017, pp. 326–335.

[Das+18] Abhishek Das, Samyak Datta, Georgia Gkioxari, Stefan Lee, Devi Parikh, and Dhruv Batra. “Embodied question answering”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*. 2018, pp. 2054–2063.

[Den+09] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. “Imagenet: A large-scale hierarchical image database”. In: *2009 IEEE conference on computer vision and pattern recognition*. Ieee. 2009, pp. 248–255.

[Den+10] Jia Deng, Alexander C Berg, Kai Li, and Li Fei-Fei. “What does classifying more than 10,000 image categories tell us?” In: *Proceedings of the European Conference on Computer Vision (ECCV)*. Springer. 2010, pp. 71–84.

[Dev+15] Jacob Devlin, Hao Cheng, Hao Fang, Saurabh Gupta, Li Deng, Xiaodong He, Geoffrey Zweig, and Margaret Mitchell. "Language Models for Image Captioning: The Quirks and What Works". In: *arXiv preprint arXiv:1505.01809* (2015).

[Dix+18] Lucas Dixon, John Li, Jeffrey Sorensen, Nithum Thain, and Lucy Vasserman. "Measuring and mitigating unintended bias in text classification". In: *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*. ACM. 2018, pp. 67–73.

[Doe+12] Carl Doersch, Saurabh Singh, Abhinav Gupta, Josef Sivic, and Alexei Efros. "What makes paris look like paris?" In: *ACM Transactions on Graphics* 31.4 (2012).

[Don+14] Jeff Donahue, Yangqing Jia, Oriol Vinyals, Judy Hoffman, Ning Zhang, Eric Tzeng, and Trevor Darrell. "Decaf: A deep convolutional activation feature for generic visual recognition". In: *International conference on machine learning*. 2014, pp. 647–655.

[Don+15] Jeffrey Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko, and Trevor Darrell. "Long-term recurrent convolutional networks for visual recognition and description". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015, pp. 2625–2634.

[Don+16] Jeff Donahue, Lisa Anne Hendricks, Marcus Rohrbach, Subhashini Venugopalan, Sergio Guadarrama, Kate Saenko, and Trevor Darrell. "Long-term recurrent convolutional networks for visual recognition and description". In: *TPAMI* (2016).

[DT05] Navneet Dalal and Bill Triggs. "Histograms of oriented gradients for human detection". In: *international Conference on computer vision & Pattern Recognition (CVPR'05)*. Vol. 1. IEEE Computer Society. 2005, pp. 886–893.

[Duy+02] Pinar Duygulu, Kobus Barnard, Joao FG de Freitas, and David A Forsyth. "Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary". In: *European conference on computer vision*. Springer. 2002, pp. 97–112.

[DV+17] Harm De Vries, Florian Strub, Sarath Chandar, Olivier Pietquin, Hugo Larochelle, and Aaron Courville. "Guesswhat?! visual object discovery through multi-modal dialogue". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2017, pp. 5503–5512.

[DVK18] Finale Doshi-Velez and Been Kim. "Considerations for evaluation and generalization in interpretable machine learning". In: *Explainable and Interpretable Models in Computer Vision and Machine Learning*. Springer, 2018, pp. 3–17.

[Dwo+12] Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard Zemel. "Fairness through awareness". In: *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference*. ACM. 2012, pp. 214–226.

[DZL17] Bo Dai, Yuqi Zhang, and Dahua Lin. "Detecting Visual Relationships with Deep Relational Networks". In: *arXiv preprint arXiv:1704.03114* (2017).

[EEH14] Eran Eidinger, Roee Enbar, and Tal Hassner. “Age and gender estimation of unfiltered faces”. In: *IEEE Transactions on Information Forensics and Security* 9.12 (2014), pp. 2170–2179.

[Ehs+18] Upol Ehsan, Brent Harrison, Larry Chan, and Mark O Riedl. “Rationalization: A neural machine translation approach to generating natural language explanations”. In: *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*. ACM. 2018, pp. 81–87.

[Fan+15] Hao Fang, Saurabh Gupta, Forrest Iandola, Rupesh K Srivastava, Li Deng, Piotr Dollár, Jianfeng Gao, Xiaodong He, Margaret Mitchell, John C Platt, et al. “From captions to visual concepts and back”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015, pp. 1473–1482.

[Far+09] Ali Farhadi, Ian Endres, Derek Hoiem, and David Forsyth. “Describing objects by their attributes”. In: *2009 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE. 2009, pp. 1778–1785.

[Far+10] Ali Farhadi, Mohsen Hejrati, Mohammad Amin Sadeghi, Peter Young, Cyrus Rashtchian, Julia Hockenmaier, and David Forsyth. “Every picture tells a story: Generating sentences from images”. In: *European conference on computer vision*. Springer. 2010, pp. 15–29.

[Fro+13] Andrea Frome, Greg Corrado, Jon Shlens, Samy Bengio, Jeffrey Dean, Marc’Aurelio Ranzato, and Tomas Mikolov. “DeViSE: A Deep Visual-Semantic Embedding Model”. In: *Advances in Neural Information Processing Systems (NIPS)*. 2013.

[FV17] Ruth C Fong and Andrea Vedaldi. “Interpretable explanations of black boxes by meaningful perturbation”. In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. 2017.

[Gao+16] Yang Gao, Oscar Beijbom, Ning Zhang, and Trevor Darrell. “Compact Bilinear Pooling”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016.

[Gao+17] Jiyang Gao, Chen Sun, Zhenheng Yang, and Ram Nevatia. “TALL: Temporal Activity Localization via Language Query”. In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)* (2017).

[GGM15] Georgia Gkioxari, Ross Girshick, and Jitendra Malik. “Contextual action recognition with r* cnn”. In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. 2015, pp. 1080–1088.

[GGVG15] Michael Gygli, Helmut Grabner, and Luc Van Gool. “Video summarization by learning submodular mixtures of objectives”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015.

[Gir+14] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. “Rich feature hierarchies for accurate object detection and semantic segmentation”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2014, pp. 580–587.

[Gor+18] Daniel Gordon, Aniruddha Kembhavi, Mohammad Rastegari, Joseph Redmon, Dieter Fox, and Ali Farhadi. “Iqa: Visual question answering in interactive environments”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018, pp. 4089–4098.

[Goy+17] Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. “Making the V in VQA matter: Elevating the role of image understanding in Visual Question Answering”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2017, pp. 6904–6913.

[GSC16] Michael Gygli, Yale Song, and Liangliang Cao. “Video2GIF: Automatic Generation of Animated GIFs from Video”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016.

[Gua+13] Sergio Guadarrama, Niveda Krishnamoorthy, Girish Malkarnenkar, Subhashini Venugopalan, Raymond Mooney, Trevor Darrell, and Kate Saenko. “YouTube2Text: Recognizing and Describing Arbitrary Activities Using Semantic Hierarchies and Zero-Shoot Recognition”. In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. 2013.

[Gup+09] Abhinav Gupta, Praveen Srinivasan, Jianbo Shi, and Larry Davis. “Understanding videos, constructing plots learning a visually grounded storyline model from annotated videos”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2009.

[GVD13] Jonathan Gordon and Benjamin Van Durme. “Reporting bias and knowledge acquisition”. In: *Proceedings of the 2013 workshop on Automated Knowledge Base Construction*. ACM. 2013, pp. 25–30.

[Har90] Stevan Harnad. “The symbol grounding problem”. In: *Physica D: Nonlinear Phenomena* 42.1-3 (1990), pp. 335–346.

[He+16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. “Deep residual learning for image recognition”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 770–778.

[Hen+16a] Lisa Anne Hendricks, Subhashini Venugopalan, Marcus Rohrbach, Raymond Mooney, Kate Saenko, and Trevor Darrell. “Deep compositional captioning: Describing novel object categories without paired training data”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016, pp. 1–10.

[Hen+16b] Lisa Anne Hendricks, Zeynep Akata, Marcus Rohrbach, Jeff Donahue, Bernt Schiele, and Trevor Darrell. “Generating Visual Explanations”. In: *ECCV*. 2016.

[Hen+17] Lisa Anne Hendricks, Oliver Wang, Eli Shechtman, Josef Sivic, Trevor Darrell, and Bryan Russell. “Localizing moments in video with natural language”. In: *Proceedings of the IEEE International Conference on Computer Vision*. 2017, pp. 5803–5812.

[Hen+18a] Lisa Anne Hendricks, Ronghang Hu, Trevor Darrell, and Zeynep Akata. “Grounding visual explanations”. In: *Proceedings of the European Conference on Computer Vision (ECCV)*. 2018, pp. 264–279.

[Hen+18b] Lisa Anne Hendricks, Oliver Wang, Eli Shechtman, Josef Sivic, Trevor Darrell, and Bryan Russell. “Localizing Moments in Video with Temporal Language”. In: *arXiv preprint arXiv:1809.01337* (2018).

[Hen+18c] Lisa Anne Hendricks, Kaylee Burns, Kate Saenko, Trevor Darrell, and Anna Rohrbach. “Women also snowboard: Overcoming bias in captioning models”. In: *European Conference on Computer Vision*. Springer. 2018, pp. 793–811.

[HL08] Mark J. Huiskes and Michael S. Lew. “The MIR Flickr Retrieval Evaluation”. In: *MIR ’08: Proceedings of the 2008 ACM International Conference on Multimedia Information Retrieval*. Vancouver, Canada: ACM, 2008.

[HM19] Drew A Hudson and Christopher D Manning. “GQA: a new dataset for compositional question answering over real-world images”. In: *arXiv preprint arXiv:1902.09506* (2019).

[HP+18] Dong Huk Park, Lisa Anne Hendricks, Zeynep Akata, Anna Rohrbach, Bernt Schiele, Trevor Darrell, and Marcus Rohrbach. “Multimodal explanations: Justifying decisions and pointing to the evidence”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018, pp. 8779–8788.

[HPS+16] Moritz Hardt, Eric Price, Nati Srebro, et al. “Equality of opportunity in supervised learning”. In: *Advances in Neural Information Processing Systems (NIPS)*. 2016, pp. 3315–3323.

[HS97] Sepp Hochreiter and Jürgen Schmidhuber. “Long short-term memory”. In: *Neural computation* 9.8 (1997), pp. 1735–1780.

[Hu+16] Ronghang Hu, Huazhe Xu, Marcus Rohrbach, Jiashi Feng, Kate Saenko, and Trevor Darrell. “Natural language object retrieval”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2016, pp. 4555–4564.

[Hu+17a] Ronghang Hu, Jacob Andreas, Marcus Rohrbach, Trevor Darrell, and Kate Saenko. “Learning to reason: End-to-end module networks for visual question answering”. In: *Proceedings of the IEEE International Conference on Computer Vision*. 2017, pp. 804–813.

[Hu+17b] Ronghang Hu, Marcus Rohrbach, Jacob Andreas, Trevor Darrell, and Kate Saenko. “Modeling Relationships in Referential Expressions with Compositional Modular Networks”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2017).

[Jan+17] Yunseok Jang, Yale Song, Youngjae Yu, Youngjin Kim, and Gunhee Kim. “TGIF-QA: Toward Spatio-Temporal Reasoning in Visual Question Answering”. In: *arXiv preprint arXiv:1704.04497* (2017).

[Jia+14] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell. “Caffe: Convolutional architecture for fast feature embedding”. In: *Proceedings of the 22nd ACM international conference on Multimedia*. ACM. 2014, pp. 675–678.

[Jia+15] Xu Jia, Efstratios Gavves, Basura Fernando, and Tinne Tuytelaars. “Guiding Long-Short Term Memory for Image Caption Generation”. In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)* (2015).

[Jia+16] Zhuolin Jiang, Yaming Wang, Larry Davis, Walt Andrews, and Viktor Rozgic. “Learning Discriminative Features via Label Consistent Neural Network”. In: *arXiv preprint arXiv:1602.01168* (2016).

[JKFF16] Justin Johnson, Andrej Karpathy, and Li Fei-Fei. “Densecap: Fully convolutional localization networks for dense captioning”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2016, pp. 4565–4574.

[JLM03] Jiwoon Jeon, Victor Lavrenko, and Raghavan Manmatha. “Automatic image annotation and retrieval using cross-media relevance models”. In: *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*. ACM. 2003, pp. 119–126.

[Joh+17a] Justin Johnson, Bharath Hariharan, Laurens van der Maaten, Li Fei-Fei, C Lawrence Zitnick, and Ross Girshick. “Clevr: A diagnostic dataset for compositional language and elementary visual reasoning”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2017, pp. 2901–2910.

[Joh+17b] Justin Johnson, Bharath Hariharan, Laurens van der Maaten, Judy Hoffman, Li Fei-Fei, C Lawrence Zitnick, and Ross Girshick. “Inferring and executing programs for visual reasoning”. In: *Proceedings of the IEEE International Conference on Computer Vision*. 2017, pp. 2989–2998.

[Joh94] W Lewis Johnson. “Agents that Learn to Explain Themselves.” In: *AAAI*. 1994, pp. 1257–1263.

[JW19] Sarthak Jain and Byron C. Wallace. “Attention is not Explanation”. In: *arXiv preprint arXiv:1902.10186v1* (2019).

[Kaz+14] Sahar Kazemzadeh, Vicente Ordonez, Mark Matten, and Tamara L Berg. “ReferItGame: Referring to Objects in Photographs of Natural Scenes.” In: *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 2014.

[KFF15] Andrej Karpathy and Li Fei-Fei. “Deep visual-semantic alignments for generating image descriptions”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015, pp. 3128–3137.

[Kim+18] Jinkyu Kim, Anna Rohrbach, Trevor Darrell, John Canny, and Zeynep Akata. “Textual explanations for self-driving vehicles”. In: *Proceedings of the European Conference on Computer Vision (ECCV)*. 2018, pp. 563–578.

[Kle+15] Benjamin Klein, Guy Lev, Gil Sadeh, and Lior Wolf. “Associating neural word embeddings with deep image representations using fisher vectors”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015.

[Kon08] Savas Konur. “An interval logic for natural language semantics.” In: *Advances in Modal Logic* 7 (2008), pp. 177–191.

[Kot+19] Satwik Kottur, José MF Moura, Devi Parikh, Dhruv Batra, and Marcus Rohrbach. “CLEVR-Dialog: A Diagnostic Dataset for Multi-Round Reasoning in Visual Dialog”. In: *arXiv preprint arXiv:1903.03166* (2019).

[Kri+13] Niveda Krishnamoorthy, Girish Malkarnenkar, Raymond J. Mooney, Kate Saenko, and Sergio Guadarrama. “Generating Natural-Language Video Descriptions Using Text-Mined Knowledge”. In: *Proceedings of the Conference on Artificial Intelligence (AAAI)*. 2013.

[Kri+17] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. “Visual genome: Connecting language and vision using crowdsourced dense image annotations”. In: *International Journal of Computer Vision* 123.1 (2017), pp. 32–73.

[KSH12] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. “Imagenet classification with deep convolutional neural networks”. In: *Advances in neural information processing systems*. 2012, pp. 1097–1105.

[KSZ14] Ryan Kiros, Ruslan Salakhutdinov, and Rich Zemel. “Multimodal neural language models”. In: *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*. 2014, pp. 595–603.

[Kul+13] Gaurav Kulkarni, Visruth Premraj, Vicente Ordonez, Sudipta Dhar, Siming Li, Yejin Choi, Alexander C Berg, and Tamara Berg. “Babytalk: Understanding and generating simple image descriptions”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* (2013).

[Lan+05] H Chad Lane, Mark G Core, Michael Van Lent, Steve Solomon, and Dave Gomboc. *Explainable artificial intelligence for training and tutoring*. Tech. rep. DTIC Document, 2005.

[Lar17] Brian N Larson. “Gender as a variable in natural-language processing: Ethical considerations”. In: (2017).

[LD02] Carmen Lacave and Francisco J Díez. “A review of explanation methods for Bayesian networks”. In: *The Knowledge Engineering Review* 17.02 (2002), pp. 107–127.

[LH05] Kenneth C Litkowski and Orin Hargraves. “The preposition project”. In: *Proceedings of the Second ACL-SIGSEM Workshop on the Linguistic Dimensions of Prepositions and their Use in Computational Linguistics Formalisms and Applications*. 2005, pp. 171–179.

[LH15] Gil Levi and Tal Hassner. “Age and gender classification using convolutional neural networks”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPR Workshops)*. 2015, pp. 34–42.

[Li+16] Yuncheng Li, Yale Song, Liangliang Cao, Joel Tetreault, Larry Goldberg, Alejandro Jaimes, and Jiebo Luo. “TGIF: A New Dataset and Benchmark on Animated GIF Description”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2016).

[Lin+14a] Dahua Lin, Sanja Fidler, Chen Kong, and Raquel Urtasun. “Visual semantic search: Retrieving videos via complex textual queries”. In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. 2014.

[Lin+14b] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. “Microsoft coco: Common objects in context”. In: *European conference on computer vision*. Springer. 2014, pp. 740–755.

[Lin91] Jianhua Lin. “Divergence measures based on the Shannon entropy”. In: *IEEE Transactions on Information theory* 37.1 (1991), pp. 145–151.

[Liu+15] Wu Liu, Tao Mei, Yongdong Zhang, Cherry Che, and Jiebo Luo. “Multi-task deep visual-semantic embedding for video thumbnail selection”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015.

[Liu+17] Siqi Liu, Zhenhai Zhu, Ning Ye, Sergio Guadarrama, and Kevin Murphy. “Improved image captioning via policy gradient optimization of spider”. In: *Proceedings of the IEEE international conference on computer vision*. 2017, pp. 873–881.

[Liu+18a] Bingbin Liu, Serena Yeung, Edward Chou, De-An Huang, Li Fei-Fei, and Juan Carlos Niebles. “Temporal Modular Networks for Retrieving Complex Compositional Activities in Videos”. In: *Proceedings of the European Conference on Computer Vision (ECCV)*. 2018, pp. 552–568.

[Liu+18b] Meng Liu, Xiang Wang, Liqiang Nie, Xiangnan He, Baoquan Chen, and Tat-Seng Chua. “Attentive moment retrieval in videos”. In: *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*. ACM. 2018, pp. 15–24.

[LNH14] C.H. Lampert, H. Nickisch, and S. Harmeling. “Attribute-Based Classification for Zero-Shot Visual Object Categorization”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*. 2014.

[Lom+12] Meghann Lomas, Robert Chevalier, Ernest Vincent Cross II, Robert Christopher Garrett, John Hoare, and Michael Kopack. "Explaining robot actions". In: *Proceedings of the seventh annual ACM/IEEE international conference on Human-Robot Interaction*. ACM. 2012, pp. 187–188.

[Lom06] Tania Lombrozo. "The structure and function of explanations". In: *Trends in Cognitive Science* 10.10 (2006).

[Lom12] T. Lombrozo. *Explanation and abductive inference.* The Oxford handbook of thinking and reasoning, 2012.

[Lu+16] Jiasen Lu, Jianwei Yang, Dhruv Batra, and Devi Parikh. "Hierarchical question-image co-attention for visual question answering". In: *Advances In Neural Information Processing Systems*. 2016, pp. 289–297.

[Lu+18] Jiasen Lu, Jianwei Yang, Dhruv Batra, and Devi Parikh. "Neural baby talk". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018, pp. 7219–7228.

[Luo+18] Ruotian Luo, Brian Price, Scott Cohen, and Gregory Shakhnarovich. "Discriminability objective for training descriptive captions". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018, pp. 6964–6974.

[MA+18] Oisin Mac Aodha, Shihan Su, Yuxin Chen, Pietro Perona, and Yisong Yue. "Teaching categories to human learners with visual explanations". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018, pp. 3820–3828.

[Mac+17] Haley MacLeod, Cynthia L. Bennett, Meredith Ringel Morris, and Edward Cutrell. "Understanding Blind People's Experiences with Computer-Generated Captions of Social Media Images". In: *Proceedings of the 2017 SIGCHI Conference on Human Factors in Computing Systems*. 2017.

[Mao+15a] Junhua Mao, Wei Xu, Yi Yang, Jiang Wang, Zhiheng Huang, and Alan Yuille. "Deep captioning with multimodal recurrent neural networks (m-rnn)". In: *Proceedings of the International Conference on Learning Representations (ICLR)*. 2015.

[Mao+15b] Junhua Mao, Wei Xu, Yi Yang, Jiang Wang, Zhiheng Huang, and Alan L. Yuille. "Learning like a Child: Fast Novel Visual Concept Learning from Sentence Descriptions of Images". In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. 2015.

[Mao+16] Junhua Mao, Jonathan Huang, Alexander Toshev, Oana Camburu, Alan L Yuille, and Kevin Murphy. "Generation and comprehension of unambiguous object descriptions". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 11–20.

[MF03] Jordi Mas and Gabriel Fernandez. "Video shot boundary detection based on color histogram". In: *Notebook Papers TRECVID2003, Gaithersburg, Maryland, NIST* (2003).

[Mik+13] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. “Distributed representations of words and phrases and their compositionality”. In: *Advances in Neural Information Processing Systems (NIPS)*. 2013, pp. 3111–3119.

[Mil16] Emiel van Miltenburg. “Stereotyping and bias in the Flickr30k dataset”. In: *Workshop on Multimodal Corpora: Computer vision and language processing*. 2016.

[Mil18] Tim Miller. “Explanation in artificial intelligence: Insights from the social sciences”. In: *Artificial Intelligence* (2018).

[Mis+16] Ishan Misra, C Lawrence Zitnick, Margaret Mitchell, and Ross Girshick. “Seeing through the human reporting bias: Visual classifiers from noisy human-centric labels”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE. 2016, pp. 2930–2939.

[MRF15] Mateusz Malinowski, Marcus Rohrbach, and Mario Fritz. “Ask your neurons: A neural-based approach to answering questions about images”. In: *Proceedings of the IEEE international conference on computer vision*. 2015, pp. 1–9.

[MSD18] Varun Manjunatha, Nirat Saini, and Larry S Davis. “Explicit Bias Discovery in Visual Question Answering Models”. In: *arXiv preprint arXiv:1811.07789* (2018).

[Mun+16] Jonghwan Mun, Paul Hongsuck Seo, Ilchae Jung, and Bohyung Han. “MarioQA: Answering Questions by Watching Gameplay Videos”. In: *arXiv preprint arXiv:1612.01669* (2016).

[Ota+16] Mayu Otani, Yuta Nakashima, Esa Rahtu, Janne Heikkilä, and Naokazu Yokoya. “Learning Joint Representations of Videos and Sentences with Web Image Search”. In: *Proceedings of the European Conference on Computer Vision Workshops (ECCV Workshops)*. 2016.

[Pac+13] Michael Pacer, Joseph Williams, Xi Chen, Tania Lombrozo, and Thomas Griffiths. “Evaluating computational models of explanation using human judgments”. In: *Proceedings of the Twenty-Ninth Conference on Uncertainty in Artificial Intelligence* (2013).

[Pan+16] Yingwei Pan, Tao Mei, Ting Yao, Houqiang Li, and Yong Rui. “Jointly modeling embedding and translation to bridge video and language”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016.

[Pap+02] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. “BLEU: a method for automatic evaluation of machine translation”. In: *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*. 2002.

[PDS18] Vitali Petsiuk, Abir Das, and Kate Saenko. “RISE: Randomized Input Sampling for Explanation of Black-box Models”. In: *arXiv preprint arXiv:1806.07421* (2018).

[Pey+17] Julia Peyre, Ivan Laptev, Cordelia Schmid, and Josef Sivic. “Weakly-supervised learning of visual relations”. In: *arXiv preprint arXiv:1707.09472* (2017).

[PG11] D. Parikh and K. Grauman. “Relative attributes”. In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. 2011.

[PH04] Ian Pratt-Hartmann. “Temporal prepositions and their logic”. In: *Temporal Representation and Reasoning, 2004. TIME 2004. Proceedings. 11th International Symposium on*. IEEE. 2004, pp. 7–8.

[Plu+15] Bryan A Plummer, Liwei Wang, Chris M Cervantes, Juan C Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. “Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models”. In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. 2015.

[Plu+17] Bryan A Plummer, Arun Mallya, Christopher M Cervantes, Julia Hockenmaier, and Svetlana Lazebnik. “Phrase Localization and Visual Relationship Detection with Comprehensive Image-Language Cues”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2017, pp. 1928–1937.

[PP14] United States. Executive Office of the President and John Podesta. *Big data: Seizing opportunities, preserving values*. White House, Executive Office of the President, 2014.

[PSM14] Jeffrey Pennington, Richard Socher, and Christopher D Manning. “Glove: Global vectors for word representation”. In: *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)* 12 (2014), pp. 1532–1543.

[PYH14] Micah Hodosh Peter Young Alice Lai and Julia Hockenmaier. “From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions”. In: *Transactions of the Association for Computational Linguistics (TACL)* 2 (2014). URL: `http://www.aclweb.org/anthology/Q/Q14/Q14-1006.pdf`.

[QPS09] Novi Quadrianto, James Petterson, and Alex J Smola. “Distribution matching for transduction”. In: *Advances in Neural Information Processing Systems (NIPS)*. 2009, pp. 1500–1508.

[QS17] Novi Quadrianto and Viktoriia Sharmanska. “Recycling privileged learning and distribution matching for fairness”. In: *Advances in Neural Information Processing Systems (NIPS)*. 2017, pp. 677–688.

[RAL18] Sainandan Ramakrishnan, Aishwarya Agrawal, and Stefan Lee. “Overcoming language priors in visual question answering with adversarial regularization”. In: *Advances in Neural Information Processing Systems*. 2018, pp. 1541–1551.

[Ram+17] Vasili Ramanishka, Abir Das, Jianming Zhang, and Kate Saenko. “Top-down visual saliency guided by captions”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Vol. 1. 2. 2017, p. 7.

[RAM18] Hee Jung Ryu, Hartwig Adam, and Margaret Mitchell. “InclusiveFaceNet: Improving Face Attribute Detection with Race and Gender Diversity”. In: *Workshop on Fairness, Accountability, and Transparency in Machine Learning (FAT/ML)*. 2018.

[Ran+16] Marc'Aurelio Ranzato, Sumit Chopra, Michael Auli, and Wojciech Zaremba. "Sequence level training with recurrent neural networks". In: *Proceedings of the International Conference on Learning Representations (ICLR)*. 2016.

[Ras+10] Cyrus Rashtchian, Peter Young, Micah Hodosh, and Julia Hockenmaier. "Collecting image annotations using Amazon's Mechanical Turk". In: *Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk*. Association for Computational Linguistics. 2010, pp. 139–147.

[Ree+16a] Scott Reed, Zeynep Akata, Xinchen Yan, Lajanugen Logeswaran, Bernt Schiele, and Honglak Lee. "Generative adversarial text to image synthesis". In: *ICML* (2016).

[Ree+16b] Scott Reed, Zeynep Akata, Honglak Lee, and Bernt Schiele. "Learning Deep Representations of Fine-Grained Visual Descriptions". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016.

[Reg+13] Michaela Regneri, Marcus Rohrbach, Dominikus Wetzel, Stefan Thater, Bernt Schiele, and Manfred Pinkal. "Grounding action descriptions in videos". In: *Transactions of the Association for Computational Linguistics (TACL)*. 2013.

[Ren+17] Steven J Rennie, Etienne Marcheret, Youssef Mroueh, Jerret Ross, and Vaibhava Goel. "Self-critical sequence training for image captioning". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2017, pp. 7008–7024.

[RHDV17] Andrew Slavin Ross, Michael C Hughes, and Finale Doshi-Velez. "Right for the right reasons: Training differentiable models by constraining their explanations". In: *arXiv preprint arXiv:1703.03717* (2017).

[Roh+10] Marcus Rohrbach, Michael Stark, György Szarvas, Iryna Gurevych, and Bernt Schiele. "What helps Where - and Why? Semantic Relatedness for Knowledge Transfer". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2010.

[Roh+13] Marcus Rohrbach, Wei Qiu, Ivan Titov, Stefan Thater, Manfred Pinkal, and Bernt Schiele. "Translating video content to natural language descriptions". In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. 2013.

[Roh+14] Anna Rohrbach, Marcus Rohrbach, Wei Qiu, Annemarie Friedrich, Manfred Pinkal, and Bernt Schiele. "Coherent multi-sentence video description with variable level of detail". In: *Proceedings of the German Confeence on Pattern Recognition (GCPR)*. 2014.

[Roh+16] Anna Rohrbach, Marcus Rohrbach, Ronghang Hu, Trevor Darrell, and Bernt Schiele. "Grounding of textual phrases in images by reconstruction". In: *Proceedings of the European Conference on Computer Vision (ECCV)* (2016).

[Roh+17a] Anna Rohrbach, Atousa Torabi, Marcus Rohrbach, Niket Tandon, Chris Pal, Hugo Larochelle, Aaron Courville, and Bernt Schiele. “Movie Description”. In: *International Journal of Computer Vision (IJCV)* (2017).

[Roh+17b] Anna Rohrbach, Makarand Tapaswi, Atousa Torabi, Tegan Maharaj, Marcus Rohrbach, Sanja Fidler Christopher Pal, and Bernt Schiele. *The Joint Video and Language Understanding Workshop: MovieQA and The Large Scale Movie Description Challenge (LSMDC)*. https://sites.google.com/site/describingmovies/lsmdc-2017. 2017.

[Roh+18] Anna Rohrbach, Lisa Anne Hendricks, Kaylee Burns, Trevor Darrell, and Kate Saenko. “Object hallucination in image captioning”. In: *arXiv preprint arXiv:1809.02156* (2018).

[RRS15] Anna Rohrbach, Marcus Rohrbach, and Bernt Schiele. “The long-short story of movie description”. In: *Proceedings of the German Confeence on Pattern Recognition (GCPR)*. 2015.

[RSG16] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. “Why should i trust you?: Explaining the predictions of any classifier”. In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM. 2016, pp. 1135–1144.

[Rus+15] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. “Imagenet large scale visual recognition challenge”. In: *International Journal of Computer Vision (IJCV)* 115.3 (2015), pp. 211–252.

[San+17] Adam Santoro, David Raposo, David G Barrett, Mateusz Malinowski, Razvan Pascanu, Peter Battaglia, and Timothy Lillicrap. “A simple neural network module for relational reasoning”. In: *Advances in neural information processing systems*. 2017, pp. 4967–4976.

[SB75] Edward H Shortliffe and Bruce G Buchanan. “A model of inexact reasoning in medicine”. In: *Mathematical biosciences* 23.3 (1975), pp. 351–379.

[SC17] Pierre Stock and Moustapha Cisse. “ConvNets and ImageNet Beyond Accuracy: Explanations, Bias Detection, Adversarial Examples and Model Criticism”. In: *arXiv preprint arXiv:1711.11443* (2017).

[Sel+17] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. “Grad-cam: Visual explanations from deep networks via gradient-based localization”. In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. 2017.

[Sen+15] Ozan Sener, Amir R Zamir, Silvio Savarese, and Ashutosh Saxena. “Unsupervised semantic parsing of video collections”. In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. 2015.

[SGS16] Aidean Sharghi, Boqing Gong, and Mubarak Shah. "Query-Focused Extractive Video Summarization". In: *Proceedings of the European Conference on Computer Vision (ECCV)*. 2016.

[She+17a] Ravi Shekhar, Sandro Pezzelle, Yauhen Klimovich, Aurélie Herbelot, Moin Nabi, Enver Sangineto, and Raffaella Bernardi. "FOIL it! Find One mismatch between Image and Language caption". In: *arXiv preprint arXiv:1705.01359* (2017).

[She+17b] Rakshith Shetty, Marcus Rohrbach, Lisa Anne Hendricks, Mario Fritz, and Bernt Schiele. "Speaking the same language: Matching machine to human captions by adversarial training". In: *Proceedings of the IEEE International Conference on Computer Vision*. 2017, pp. 4135–4144.

[Sig+16] Gunnar A Sigurdsson, Gül Varol, Xiaolong Wang, Ali Farhadi, Ivan Laptev, and Abhinav Gupta. "Hollywood in Homes: Crowdsourcing Data Collection for Activity Understanding". In: *Proceedings of the European Conference on Computer Vision (ECCV)*. 2016.

[Sis01] Jeffrey Mark Siskind. "Grounding the lexical semantics of verbs in visual perception using force dynamics and event logic". In: *Journal of artificial intelligence research* 15 (2001), pp. 31–90.

[Soc+13] Richard Socher, Milind Ganjoo, Christopher D Manning, and Andrew Ng. "Zero-Shot Learning Through Cross-Modal Transfer". In: *Advances in Neural Information Processing Systems (NIPS)*. 2013.

[Soc+14] Richard Socher, Andrej Karpathy, Quoc V Le, Christopher D Manning, and Andrew Y Ng. "Grounded compositional semantics for finding and describing images with sentences". In: *Transactions of the Association for Computational Linguistics (TACL)* (2014).

[Son+15] Yale Song, Jordi Vallmitjana, Amanda Stent, and Alejandro Jaimes. "Tvsum: Summarizing web videos using titles". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015.

[SVL14] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. "Sequence to sequence learning with neural networks". In: *Advances in neural information processing systems*. 2014, pp. 3104–3112.

[SZ15] Karen Simonyan and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition". In: *Proceedings of the International Conference on Learning Representations (ICLR)* (2015).

[Sze+16] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. "Rethinking the inception architecture for computer vision". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2016, pp. 2818–2826.

[Tan+18] Sarah Tan, Rich Caruana, Giles Hooker, and Yin Lou. “Detecting Bias in Black-Box Models Using Transparent Model Distillation”. In: *AAAI/ACM Conference on Artificial Intelligence, Ethics, and Society*. 2018.

[Tap+16] Makarand Tapaswi, Yukun Zhu, Rainer Stiefelhagen, Antonio Torralba, Raquel Urtasun, and Sanja Fidler. “Movieqa: Understanding stories in movies through question-answering”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016.

[Tho+14] Jesse Thomason, Subhashini Venugopalan, Sergio Guadarrama, Kate Saenko, and Raymond J. Mooney. “Integrating Language and Vision to Generate Natural Language Descriptions of Videos in the Wild”. In: *Proceedings of the International Conference on Computational Linguistics (COLING)*. 2014.

[Tho+15] Bart Thomee, David A Shamma, Gerald Friedland, Benjamin Elizalde, Karl Ni, Douglas Poland, Damian Borth, and Li-Jia Li. “The new data and new challenges in multimedia research”. In: *arXiv preprint arXiv:1503.01817* (2015).

[Tor02] Antonio Torralba. “Contextual modulation of target saliency”. In: *Advances in Neural Information Processing Systems (NIPS)*. 2002, pp. 1303–1310.

[Tou+03] Kristina Toutanova, Dan Klein, Christopher D Manning, and Yoram Singer. “Feature-rich part-of-speech tagging with a cyclic dependency network”. In: *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*. Association for Computational Linguistics. 2003, pp. 173–180.

[TP12] Bart Thomee and Adrian Popescu. “Overview of the ImageCLEF 2012 Flickr Photo Annotation and Retrieval Task.” In: *CLEF (Online Working Notes/Labs/Workshop)*. Vol. 12. 2012.

[TR09] Stefanie Tellex and Deb Roy. “Towards surveillance video search by natural language query”. In: *ACM*. 2009.

[Tra+15] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. “Learning spatiotemporal features with 3d convolutional networks”. In: *Proceedings of the IEEE international conference on computer vision*. 2015, pp. 4489–4497.

[TS01] Antonio Torralba and Pawan Sinha. “Statistical context priming for object detection”. In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. Vol. 1. IEEE. 2001, pp. 763–770.

[TS81] Randy L Teach and Edward H Shortliffe. “An analysis of physician attitudes regarding computer-based clinical consultation systems”. In: *Use and impact of computers in clinical medicine*. Springer, 1981, pp. 68–85.

[TTS16] Atousa Torabi, Niket Tandon, and Leonid Sigal. “Learning Language-Visual Embedding for Movie Understanding with Natural-Language”. In: *arXiv preprint arXiv:1609.08124* (2016).

[Tze+15] Eric Tzeng, Judy Hoffman, Trevor Darrell, and Kate Saenko. “Simultaneous deep transfer across domains and tasks”. In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. IEEE. 2015, pp. 4068–4076.

[Ved+17] Ramakrishna Vedantam, Samy Bengio, Kevin Murphy, Devi Parikh, and Gal Chechik. “Context-aware Captions from Context-agnostic Supervision”. In: *arXiv preprint arXiv:1701.02870* (2017).

[Ven+15a] Subhashini Venugopalan, Marcus Rohrbach, Jeffrey Donahue, Raymond Mooney, Trevor Darrell, and Kate Saenko. “Sequence to sequence-video to text”. In: *Proceedings of the IEEE international conference on computer vision*. 2015, pp. 4534–4542.

[Ven+15b] Subhashini Venugopalan, Huijuan Xu, Jeff Donahue, Marcus Rohrbach, Raymond Mooney, and Kate Saenko. “Translating Videos to Natural Language Using Deep Recurrent Neural Networks”. In: *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*. 2015.

[Ven+17] Subhashini Venugopalan, Lisa Anne Hendricks, Marcus Rohrbach, Raymond Mooney, Trevor Darrell, and Kate Saenko. “Captioning images with diverse objects”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017, pp. 5753–5761.

[Vin+15] Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. “Show and tell: A neural image caption generator”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015, pp. 3156–3164.

[VLFM04] Michael Van Lent, William Fisher, and Michael Mancuso. “An explainable artificial intelligence system for small-unit tactical behavior”. In: *Proceedings of the National Conference on Artificial Intelligence*. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999. 2004, pp. 900–907.

[VLZP15] Ramakrishna Vedantam, C Lawrence Zitnick, and Devi Parikh. “Cider: Consensus-based image description evaluation”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2015, pp. 4566–4575.

[VZ05] Manik Varma and Andrew Zisserman. “A statistical approach to texture classification from single images”. In: *International journal of computer vision* 62.1-2 (2005), pp. 61–81.

[Wah+11] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. *The Caltech-UCSD Birds-200-2011 Dataset*. Tech. rep. CNS-TR-2011-001. California Institute of Technology, 2011.

[Wal+17] Caren M Walker, Tania Lombrozo, Joseph J Williams, Anna N Rafferty, and Alison Gopnik. “Explaining constrains causal learning in childhood”. In: *Child development* 88.1 (2017), pp. 229–246.

[Wan+16] Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. "Temporal segment networks: towards good practices for deep action recognition". In: *Proceedings of the European Conference on Computer Vision (ECCV)*. 2016.

[Wil92] Ronald J. Williams. "Simple statistical gradient-following algorithms for connectionist reinforcement learning". In: *Machine Learning* (1992).

[WL13] Joseph J Williams and Tania Lombrozo. "Explanation and prior knowledge interact to guide learning". In: *Cognitive psychology* 66.1 (2013), pp. 55–84.

[WM18] Jialin Wu and Raymond J Mooney. "Faithful Multimodal Explanation for Visual Question Answering". In: *arXiv preprint arXiv:1809.02805* (2018).

[Wu+16] Qi Wu, Chunhua Shen, Lingqiao Liu, Anthony Dick, and Anton Van Den Hengel. "What value do explicit high level concepts have in vision to language problems?" In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016, pp. 203–212.

[Wu+17] Qi Wu, Damien Teney, Peng Wang, Chunhua Shen, Anthony Dick, and Anton van den Hengel. "Visual question answering: A survey of methods and datasets". In: *Computer Vision and Image Understanding* 163 (2017), pp. 21–40.

[Wu+18] Yu Wu, Linchao Zhu, Lu Jiang, and Yi Yang. "Decoupled Novel Object Captioner". In: *Proceedings of the 2018 ACM on Multimedia Conference (ACM MM)*. 2018.

[Xu+15a] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. "Show, attend and tell: Neural image caption generation with visual attention". In: *International conference on machine learning*. 2015, pp. 2048–2057.

[Xu+15b] Ran Xu, Caiming Xiong, Wei Chen, and Jason J Corso. "Jointly Modeling Deep Video and Compositional Text to Bridge Vision and Language in a Unified Framework." In: *Proceedings of the Conference on Artificial Intelligence (AAAI)*. 2015.

[Xu+16] Jun Xu, Tao Mei, Ting Yao, and Yong Rui. "Msr-vtt: A large video description dataset for bridging video and language". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016.

[Yan+15] Huan Yang, Baoyuan Wang, Stephen Lin, David Wipf, Minyi Guo, and Baining Guo. "Unsupervised Extraction of Video Highlights Via Robust Recurrent Autoencoders". In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. 2015.

[Yao+17] Ting Yao, Yingwei Pan, Yehao Li, and Tao Mei. "Incorporating copying mechanism in image captioning for learning novel objects". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2017, pp. 6580–6588.

[Yao+18] Ting Yao, Yingwei Pan, Yehao Li, and Tao Mei. "Exploring visual relationship for image captioning". In: *Proceedings of the European Conference on Computer Vision (ECCV)*. 2018, pp. 684–699.

[Yeu+16] Serena Yeung, Olga Russakovsky, Ning Jin, Mykhaylo Andriluka, Greg Mori, and Li Fei-Fei. "Every Moment Counts: Dense Detailed Labeling of Actions in Complex Videos". In: *CVPR*. 2016.

[YFFF14] Serena Yeung, Alireza Fathi, and Li Fei-Fei. "Videoset: Video summary evaluation through text". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPR Workshops)*. 2014.

[YMR16] Ting Yao, Tao Mei, and Yong Rui. "Highlight Detection with Pairwise Deep Ranking for First-Person Video Summarization". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016.

[Yos+14] Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. "How transferable are features in deep neural networks?" In: *Advances in Neural Information Processing Systems (NIPS)*. 2014, pp. 3320–3328.

[You+14] Peter Young, Alice Lai, Micah Hodosh, and Julia Hockenmaier. "From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions". In: *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)* 2 (2014), pp. 67–78.

[YS13] Haonan Yu and Jeffrey Mark Siskind. "Grounded Language Learning from Video Described with Sentences." In: *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*. 2013.

[Yu+15a] Haonan Yu, N Siddharth, Andrei Barbu, and Jeffrey Mark Siskind. "A compositional framework for grounding language inference, generation, and acquisition in video". In: *Journal of Artificial Intelligence Research* 52 (2015), pp. 601–713.

[Yu+15b] Haonan Yu, Jiang Wang, Zhiheng Huang, Yi Yang, and Wei Xu. "Video paragraph captioning using hierarchical recurrent neural networks". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015.

[YY06] Xiang Yan and Ling Yan. "Gender Classification of Weblog Authors." In: *AAAI Spring Symposium: Computational Approaches to Analyzing Weblogs*. Palo Alto, CA. 2006, pp. 228–230.

[ZF14] Matthew D. Zeiler and Rob Fergus. "Visualizing and understanding convolutional networks". In: *Proceedings of the European Conference on Computer Vision (ECCV)*. Springer. 2014, pp. 818–833.

[Zha+15] Xu Zhang, Felix Xinnan Yu, Shih-Fu Chang, and Shengjin Wang. "Deep transfer network: Unsupervised domain adaptation". In: *arXiv preprint arXiv:1503.00591* (2015).

[Zha+16a] Jianming Zhang, Zhe Lin, Jonathan Brandt, Xiaohui Shen, and Stan Sclaroff. "Top-down neural attention by excitation backprop". In: *Proceedings of the European Conference on Computer Vision (ECCV)*. Springer. 2016, pp. 543–559.

[Zha+16b] Kaipeng Zhang, Lianzhi Tan, Zhifeng Li, and Yu Qiao. "Gender and smile classification using deep convolutional neural networks". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPR Workshops)*. 2016, pp. 34–38.

[Zha+17] Jieyu Zhao, Tianlu Wang, Mark Yatskar, Vicente Ordonez, and Kai-Wei Chang. "Men Also Like Shopping: Reducing Gender Bias Amplification using Corpus-level Constraints". In: *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 2017.

[Zha+18] Da Zhang, Xiyang Dai, Xin Wang, Yuan-Fang Wang, and Larry S Davis. "MAN: Moment Alignment Network for Natural Language Moment Retrieval via Iterative Graph Adjustment". In: *arXiv preprint arXiv:1812.00087* (2018).

[Zho+18] Luowei Zhou, Yannis Kalantidis, Xinlei Chen, Jason J Corso, and Marcus Rohrbach. "Grounded Video Description". In: *arXiv preprint arXiv:1812.06587* (2018).

[ZHS18] Bowen Zhang, Hexiang Hu, and Fei Sha. "Cross-Modal and Hierarchical Modeling of Video and Text". In: *Proceedings of the European Conference on Computer Vision (ECCV)*. 2018, pp. 374–390.

[Zin+17] Luisa M. Zintgraf, Taco S. Cohen, Tameem Adel, and Max Welling. "Visualizing deep neural network decisions: Prediction difference analysis". In: *Proceedings of the International Conference on Learning Representations (ICLR)*. 2017.

[ZLM18] Brian Hu Zhang, Blake Lemoine, and Margaret Mitchell. "Mitigating Unwanted Biases with Adversarial Learning". In: *AAAI/ACM Conference on Artificial Intelligence, Ethics, and Society (AIES)*. 2018.

[ZNWZ18] Quanshi Zhang, Ying Nian Wu, and Song-Chun Zhu. "Interpretable convolutional neural networks". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018, pp. 8827–8836.

[ZSC17] Xiaoshi Zhong, Aixin Sun, and Erik Cambria. "Time expression analysis and recognition using syntactic token types and general heuristic rules". In: *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Vol. 1. 2017, pp. 420–429.