MULTIMODAL KNOWLEDGE INTEGRATION FOR OBJECT DETECTION AND VISUAL REASONING

by

Keren Ye

M.S. in Control Science, Beihang University, 2011

B.S. in Computer Science, Beihang University, 2008

Submitted to the Graduate Faculty of

the Department of Computer Science in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy

University of Pittsburgh

2021

UNIVERSITY OF PITTSBURGH

DEPARTMENT OF COMPUTER SCIENCE

This dissertation was presented

by

Keren Ye

It was defended on

July 8th 2021

and approved by

Adriana Kovashka, Department of Computer Science

Diane Litman, Department of Computer Science

Milos Hauskrecht, Department of Computer Science

Daqing He, Department of Informatics and Networked Systems

Seong Jae Hwang, Department of Computer Science

MULTIMODAL KNOWLEDGE INTEGRATION FOR OBJECT DETECTION AND VISUAL REASONING

Keren Ye, PhD

University of Pittsburgh, 2021

We humans still perceive and reason in a different way than artificial intelligence models. We witness, we listen, we touch, we understand the world via multi-modal sensing, while machine models rely only on a single or a few modalities and ignore abundant information. In this thesis, we explore techniques for reducing the perception gap between machines and humans and focus on two families of tasks, reasoning and detection. First, we incorporate information from text, audio, motion, external knowledge bases, for training computer vision models. We find that data inputs from more extensive channels provide complementary information to improve models. Second, we study how multimodal inputs can be fully utilized. We argue that most existing deep learning methods are prone to pay too large attention to shallow patterns in the input features, which causes the resulting models to be biased. We propose robust training to overcome the issue. Third, we extend the benefits of multi-modal information to the supervision signals instead of the inputs, by learning a weakly supervised detection model from the natural supervision of textual captions or audio narrations. With the help of NLP constituency parsing, it is possible to extract structural knowledges from the captions and narrations, hence determines the entities and relations of visual objects.

keywords weakly supervised learning, object detection, scene graphs generation, cross-modal retrieval, multi-modal learning, advertisements, external knowledge, vision and language, representation learning, question answering.

Table of Contents

List of Tables

List of Figures

Preface

Before presenting the thesis work, I wish to acknowledge the help from all the people. First of all, I would like to thank my parents and all my friends, who support and encourage me to pursue a doctorate in the US. As I remember, studying abroad was a hard decision. When I look back, I find the 6-years of effort worthy and are a great treasure for my life experience.

I would like to express my deepest appreciation to my advisor Dr. Adriana Kovashka, who provides the most frequent help for my study and research. Her knowledge regarding academics, her experience in artificial intelligence, her enthusiasm for research, her interests in the area, are all the stimuli inspiring me to study, explore, and research. Especially, she is the only one who provides extremely detailed feedbacks on my paper drafts and research projects. The feedbacks and the comments from her help me to improve myself, to become a better researcher. Without her extensive and comprehensive mentorship, I cannot achieve today's academic accomplishments.

Then, I would like to extend my deepest gratitude to my committee members, Dr. Diane Litman, Dr. Milos Hauskrecht, Dr. Daqing He, and Dr. Seong Jae Hwang, for providing useful suggestions and guidance through the process. Their every comment and suggestion to the thesis do contribute to making it better.

I am also grateful to all my labmates and co-authors: Nils Murrugarra-Llerena, Christopher Thomas, Mingda Zhang, Narges Honarvar Nazari, Mesut Erhan Unal, Tristan Maidment, Ahmad Diab, Meiqi Guo, Zhexiong Liu, Zaeem Hussain, Xiaozhong Zhang, Zuha Agha, Nathan Ong, Kyle Buettner, Wei Li, Danfeng Qin, Jesse Berent, James Hahn, Rebecca Hwa, Mark Sandler, Menglong Zhu, Andrew Howard, Marco Fornoni. Our discussions regarding the research projects inspire new ideas and produce valuable research accomplishments.

I wish to thank my internship hosts and colleagues from Google: Dmitry Kalenichenko, Florian Schroff, Menglong Zhu, Ting Liu, Jesse Berent, Wei Li, Danfeng Qin, Marco Fornoni, Mark Sandler. They hosted internships for me in Los Angeles, US, Zurich, CH, and Paris, FR. These internships provide me a chance to travel around the world and let me know how the research works are done in the top industrial company.

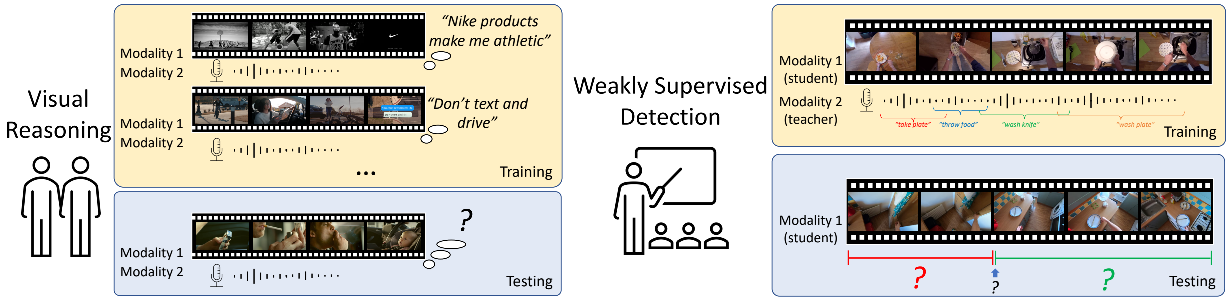
Finally, I wish to thank all the faculty and staff from the Department of Computer Science, University of Pittsburgh. Their efforts to improve both the hardware and software in the department thus provided me a happy time here.

1.0 Introduction

Developing an intelligent system behaving like a human is the mainstream study of artificial intelligence in the past decades. For example, supervised learning can be generalized as learning a function that maps an input to an output, where the golden-standard of the output is the supervision that we expect the machine to mimic. Early machine learning methods use manually extracted features for the input, hence the machine's perception of the world is through human efforts. The advent of the deep neural networks replaced the human roles in feature extraction through end-to-end training. CNN obtains information from pixels, RNN and transformer models discover patterns from tokens, all without manually specifying the feature rules. In these ways, machines sense the world more similar to human beings, and humans are partially freed from labor-intensive feature extraction works.

However, the perception gap between machines and humans still exists. The neural network models did not change the fact that most models only rely on the two conventional modalities - image and text. For example, the core tasks of computer vision (namely classification, detection, segmentation) all only use the image features. Also, the vision-and-language tasks such as image captioning, image-text matching, and visual question answering, only use the pairing of image and text. In comparison, humans perceive and reason in a different way. We witness, we listen, we touch, we understand the world via multi-modal sensing, while the models ignore abundant information in the wild. Thus, natural cues such as audio, speech, motion, scene depth are all potentially helpful and need to be explored.

More specifically, the perception gap lies in the inputs, as well as the supervised signals. For the *inputs*, models usually do not know which modalities are useful for prediction and do not know how to encode them. For example, TextVQA [233] shows that embedded texts are informative for understanding images, yet previous models for the task do not consider them as a primary modality. For the use of *multimodal supervision* that is free and abundant on the internet (videos, images paired with comments, etc.), it is not the mainstream, so institutes and companies still hire people to provide paired annotations instead. However, due to the big data era, we believe that using the weakly paired multimodal annotations has



Chapter 3, 4, 5: using multiple modalities for ads understanding

Chapter 7, 8: using captions to learn object detection models
Chapter 9: using audio narrations to learn action detection models

Figure 1: Visual reasoning v.s. Weakly supervised detection. Visual reasoning uses the multimodal features at both training and testing time, while weakly supervised detection only uses the multimodal signals for training. At testing time, the detection model has to figure out the location of visuals without the multimodal signals.

the potential which merits more attention.

To cope with the perception gap mentioned above, we incorporate both *richer features* and *richer supervision* from multimodal information. First, we propose multimodal knowledge integration for visual reasoning, to analyze images/videos with implicit persuasive intent and answer questions about images/videos. We study how to encode and fuse multimodal representations, as well as how to use them efficiently. For this part, we focus on image/video advertisements' understanding in that ads are usually well-designed to incorporate complicated and informative features from embedded texts, audio, motion, speech, and so on. Since the goal of ad design is even to enforce humans to think, the task is challenging for machine models — without fully perceiving the hints like humans, the models can hardly understand the true meanings of ads. Then, we extend the idea of using multimodal cues to conventional images and videos and study the cues that potentially serve as weakly supervised signals to localize visual objects or actions. Specifically, we attempt to learn detection models from them. As compared to multimodal knowledge integration, this part explores to use multimodal signals as supervision instead of inputs, and it targets to harvest better and more concrete vision detection models (see Fig. 1 for the differences).

1.1 Challenges

Multimodal information is not guaranteed to help. First, multimodal features and external knowledge can contain noises in practical scenarios. These bring complexity to the modeling. For example, the speech in videos often contains unrelated content (even incorrectly recognized), such as mentioning the time, discussing the popular sports events during the season, or just presenting simple greetings. The knowledge that may be retrieved from a general-purpose knowledge base is even noisier in that the external retrieval process usually considers no semantics, or it considers semantics beyond the studied domain. For example, WWF has ambiguous meanings such as World Wide Fund for Nature, The Working Women's Forum, etc., and the retrieval process just returns all of them or a random one, which may confuse the models. Therefore, without properly dealing with noises from multimodal features and external knowledge, one can hardly observe improved performance.

Then, even with clean inputs, generalizing a robust model is not easy. A well-known issue is the overfitting to shallow patterns. For example, in VQA [12], many models learn to answer three for the counting questions, without considering the visual. The reason is the dataset biases — models generalize from biased dataset, hence are not able to really reason. Thus, only when we are knowledgable regarding the data and understand the utilization of the reasoning evidence, we can train a robust model that works in any situation, rather than focusing on unreliable evidence.

Finally, properly incorporating multimodal supervision is complicated. For example, utilizing text captions for object detection involves two primary challenges: reporting biases and grounding issues. For the former, the multimodal cues may be complementary instead of redundant. So, the supervisions may not cover all the objects in the images, i.e., they choose NOT to mention some visual instances. In this case, models need to have some mechanism to correct the biases. For the latter, although the visual and the supervision signals have overlapped, determining the proper visual objects (or video actions) is still unclear since some concepts such as the zoo, paveway may not have a clear visual boundary and can not be treated as an "instance". If a model tries to localize these abstract concepts, it may suffer from a performance drop. So, models have to choose the visually concrete instances to be matched to the labels. In video action detection, using audio narrations needs to resolve the ambiguity in which the temporal boundary between consecutive actions is incorrectly annotated. Methods hence need to model the boundary uncertainty to better separate and localize action instances.

1.2 Research Statements

We divide the focus of the thesis into two primary parts and propose five hypotheses regarding them (formally introduced in Tab. 1). On the one hand, we are interested in the *multimodal knowledge integration for visual reasoning.* Through training visual reasoning models with/without extra features, we want to verify if multimodal features and external knowledge are helpful to understand images/videos with implicit persuasive intent such as visual advertisements (H1). We also expect to tackle the unreliable evidence for better model reasoning, thus achieving a generalizable and robust model. Such unreliable evidence includes but not limited to variants of dataset biases, overfitting to the shallow pattern (H2), noises in the labels, and the underutilize of supervisions (H3). On the other hand, we are eager to see if *multimodal cues can be used for localizing objects in images and actions in videos.* We investigate whether the text captions are strong enough for training an object detection model that recognizes and localizes visual objects. We investigate whether the text captions are strong enough for training an object detection model that recognizes and localizes visual objects. We explicitly deal with noises in the text supervision to learn the object detection models (H3). Then, we further transfer knowledge such as properties and relations from the text to visual models to utilize the caption supervision fully, and to infer more reliable evidence from the supervision, for localization (H4). Finally, we investigate the impact of audio and motion features, both for detection and reasoning tasks (H5).

Table 1: Hypotheses in the thesis.

Primary	#	Hypotheses
Multimodal Cues for Visual Reasoning	H1	Multimodal features help to understand images/videos with implicit persuasive intent, such as visual advertisements.
	H2	Text features can be unreliable if not modeled appropriately.
Multimodal Cues for Localization	H3	Text supervision contains noise, but can be used to localize visual objects in space, if modeled properly.
	H4	Text supervision provides contexts regarding visual objects, they are reliable cues for disambiguating entities and relations.
	H5	Noisy audio narrations as a multimodal signal can be modeled to localize video actions in temporal domain.

1.3 Outline of the Chapters

We show in Tab. 2 all the chapters in the thesis, under the two primary tasks using multimodal knowledge: visual reasoning which uses multimodal inputs, and weakly object/action detection which uses multimodal supervision. For each task (visual reasoning or detection), we first design basic models that can *use the information from multiple channels*: in Chapter 3, 4, we design simple models for understanding the implicit intent in image/video ads. Chapter 3 matches ads to the statements best describing them, while Chapter 4 predicts the temporal dynamics, i.e. the sentiment and its intensity change. In Chapter 7 and Chapter 9, we train image and video detection models respectively, using multimodal signals as supervision. We ***preview the multi-modalities*** used in the mentioned chapters:

- Chapter 3 considers in the image model the visual symbol regions, embedded text slogans, knowledge from symbols/visual objects; it also uses in the video model the video frames, as well as the narrations parsed from speech-to-text API.
- Chapter 4 base on multiple frame-level features to provide the prediction. It uses the

place the story happens, the visual objects presented, the actors' facial expressions, the magnitude of the motion changes, the estimated shot boundary, and the sound loudness.

- Chapter 7 learns image object detectors from the paired captions of the images. It extracts the entities mentioned in the texts as the image-level labels.
- Chapter 9 learns video action detectors from narration audios, considering the uncertain boundaries between consecutive actions.

After presenting the fundamental models using multimodal cues as inputs or supervision, we progressively consider a more practical issue when using the multiple modalities — noise. Though the concept seems abstract, we concretely ***define and study the noise*** in the following chapters:

- Chapter 5 integrates external knowledge for Chapter 3. However, the used text-query-guided retrieval process is unaware of the image, thus may incur noisy knowledge entries irrelevant to the ad. Besides, the query in the retrieved paragraph can be directly connected to the statements to be ranked, causing a "shortcut effect" that hinders models from true reasoning (models learn to utilize shortcuts, hence get "lazy" to uncover the underlying "true" association). The former issue is caused by a noisy external random process, and it *hinders model inference*, while the latter is caused by the biased data distribution which is *detrimental for training.* Both issues bring obstacles to learning a robust model.
- Chapter 6 investigates the shortcut effects same as in Chapter 5, but on a more general VQA dataset.
- Chapter 7 tackles the noise in the descriptive captions paired to the images. In comparison to Chapter 5, 6, the goal is to learn detection models from multimodal supervision rather than using multimodal inputs for visual reasoning. We break down this type of noise in the annotation process a bit more. On the one hand, the caption may or may not mention the objects. Thus the *quality of image-caption pairs* are different, and some may be more useful than others for learning an object detection model. By filtering out complementary or irrelevant image-caption pairs, models may potentially be improved. On the other hand, *extracting object categories* from good-quality captions incurs noise. Because some objects are always implied in the captions (e.g., "pans", "bowls" in the

"kitchen" scene, "ties" worn by "professionals") and are not explicitly mentioned, the naive way of extracting labels (e.g., lexical matching) may introduce false negatives.

- Chapter 8 further extracts more reliable supervision than Chapter 7. It extracts a holistic representation — a text graph from the caption. Thus the matching of instances, which additionally depends on the relation context, is more accurate than Chapter 7. It avoids the noisy instance labeling that associates a text entity to a random related region (e.g., connecting a "girl-in-hat" to the one without wearing a hat in the same image, since lacking the context of "-in-hat").
- Chapter 9 models the *uncertainty in the video clip* to use the noisy audio narration supervision. In the audio narrations, the start time of each narration annotation is imprecise. The end time can only be assumed to be before the start time of the next annotation. Thus whether a video frame belongs to the previous action, next action, or no action is uncertain. We build a model to capture the uncertainty.

Table 2: Published papers with regard to the thesis topics.

Primary	Secondary	Chapters	Conference / Journal	H1	H2	H3	H4	H5
Multimodal knowledge integration for visual reasoning	Using text, audio, external knowledge, etc.	Chapter 3 - ADVISE: Symbolism and External Knowledge for Decoding Advertisements	ECCV18 [294] TPAMI19 [297]	✓		✓		
		Chapter 4 - Story Understanding in Video Advertisements	BMVC18 [293]	✓				✓
	Robustly utilizing multimodal features	Chapter 5 - Breaking Shortcuts by Masking for Robust Visual Reasoning	WACV21 [298]	✓	✓			
		Chapter 6 - A Case Study of the Shortcut Effects in Visual Commonsense Reasoning	AAAI21 [295]		✓			
Multimodal cues for localizing objects and actions	Using text captions, audio narrations as supervision.	Chapter 7 - Cap2Det: Learning to Amplify Weak Caption Supervision for Object Detection	ICCV19 [299] Submitted to TPAMI			✓		
		Chapter 9 - Action Detection through Audio Narration Supervision	TBD					✓
	Robustly utilizing multimodal supervision	Chapter 8 - Linguistic Structures as Weak Supervision for Visual Scene Graph Generation	CVPR21 [296]			✓	✓	

1.4 Primary Tasks

In this section, we introduce the two primary tasks in the thesis. We work on visual reasoning initially, then find the multimodal cues are useful for localization, thus dedicate our later studies to object/action detection. The inner connection between the two is that detection provides the fundamental regarding an image/video while visual reasoning is the upper structure mimicking human thinking. For the *multimodal knowledge integration for visual reasoning*, we focus on image/video ads understanding (Sec. 1.4.1) because ads are usually well-designed to incorporate complicated and informative features from embedded texts, audio, motion, speech, and so on. Thus, they are suitable for testing model reasoning. For *localizing visual instances from multimodal cues*, we focus on weakly supervised object/action detection (Sec. 1.4.2) since learning from the enviroment (multimodal supervision) to localize and name instances is a fundamental problem in vision, and it has more significant future impacts for applications such as navigation, robotic, self-driving etc.

1.4.1 Ads Understanding

Visual media are informative, but they are also manipulative, intentionally or unintentionally [85, 181, 143, 217, 304]. Targeted campaigns to change public opinions on matters with economic and social impact have been effective [287, 43]. Well-created ads gain great popularity and are seen by many, thus entering our common consciousness [193]. The public response to political images has caused policy changes as well as major governmental decisions on issues such as war involvement and admitting refugees [18, 279].

Despite the importance of the persuasive nature of visual media, there is a scarcity of computer vision approaches to understand visual rhetoric. While we have made impressive progress on inferring the explicit content in the media (e.g. objects, scenes, actions), the implicit nuances of the media have been overlooked, partly due to the significant challenges that this task poses. Sometimes the message of an image is simple, and can be inferred from body language, as in the “We can do it” ad (A) in Fig. 2. Other images convey more complex or clever messages, whose decoding relies on human visual recognition (including generalization), association, and reasoning capabilities. For example, in Fig 2 (B), one might infer that because the eggplant and pencil form the same object, the pencil gives a very real, *natural* eggplant color, as in Fig. 2 (B). In Fig (C), one might conclude that Burger King burgers are delicious, since even employees from competitor restaurants (McDonalds) secretly buy them. In Fig (D), lungs *symbolize breathing and by extension, life.* However, a human first has to recognize the groups of trees as lungs, which might be difficult for a computer vision system to do, due to the atypical texture. In (E), the viewer has to infer that the woman went on vacation from the fact that she is carrying a suitcase, and then surmise that she is carrying dead animals from the blood trailing behind her suitcase. A human knows this because she *associates blood with injury or death.* These are just a few examples of how ads use different types of *visual rhetoric* to convey their message, namely: association and symbolism, common-sense reasoning, and recognition of non-photorealistic objects. Understanding advertisements automatically requires decoding this rhetoric.

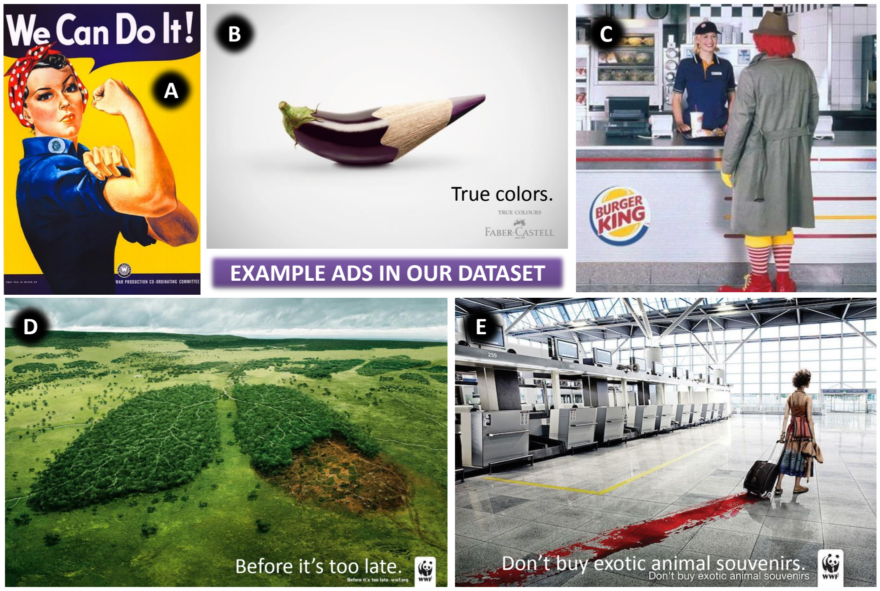

We Can Do It!
A
B
True colors.
TRUE COLOURS
FABER-CASTELL
EXAMPLE ADS IN OUR DATASET
C
BURGER KING
D
Before it's too late.
WWF
E
Don't buy exotic animal souvenirs.
Don't buy exotic animal souvenirs
WWF

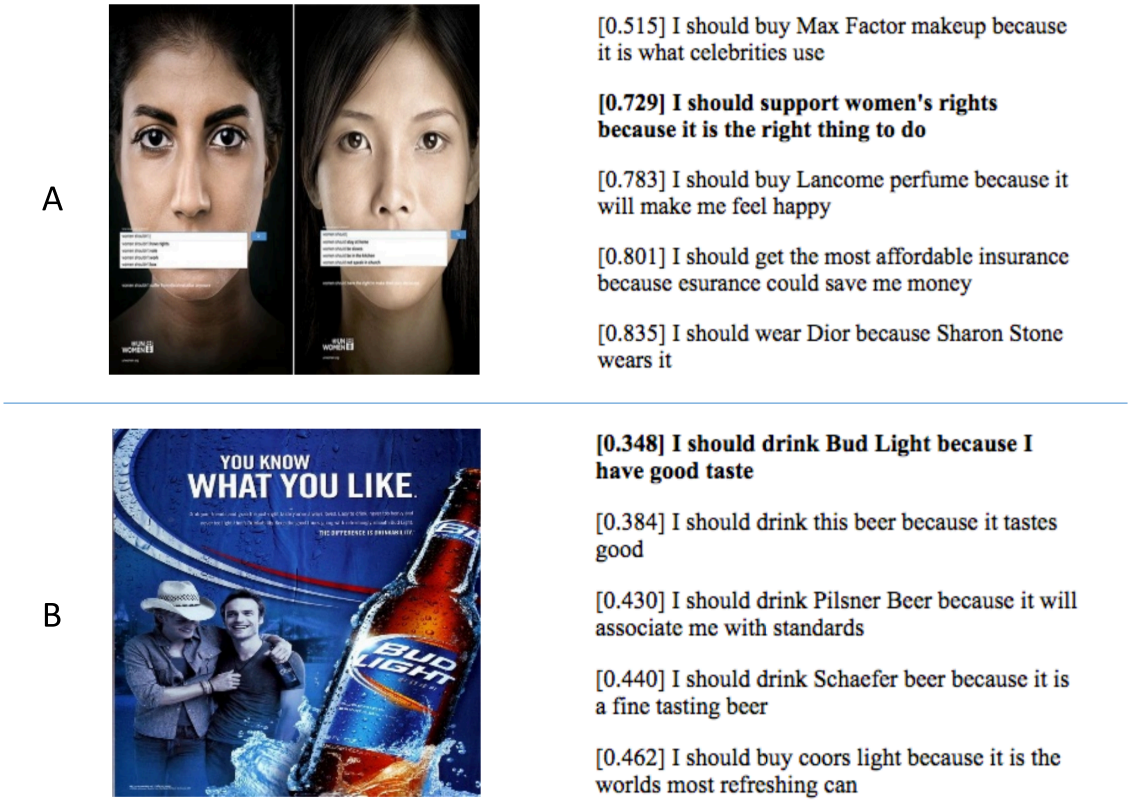
Figure 2: Example advertisements from our dataset that require challenging visual recognition and reasoning. Despite the potential applications of understanding the messages of ads, this problem has not been tackled in computer vision.

In this thesis, Chapter 3, 5 consider the same ads understanding task: given an image and several statements, the system must identify the correct statement to pair with the ad. For example, the system in Fig. 3 made an incorrect top-1 prediction for Fig. 3 (A) but a correct prediction for Fig. 3 (B). We evaluate the systems using more metrics such as precision, recall, rank, etc., which shall be introduced in the chapters. Chapter 4 also studies the ads but attempts to understand the sentiments of the video story (multi-label classification), as well as to predict the sentiment peaks (binary classification).

Besides ads, there are other datasets that also require high-level reasoning, such as visual question answering and visual commonsense reasoning, which perceive the features of an image and provide natural language responses regarding the visual contents. We study the Visual Common Reasoning (VCR) dataset in Chapter 6. The difference to the ads understanding is that a natural language question is also provided as input for each image. Thus, models are required to rank the answer options based on both the image and question. We validate that our noise-dealing method applies to the ads dataset and also this more general VCR dataset.

1.4.2 Object and Action Detection

Unlike visual reasoning, which is a high-level sensing and thinking problem, localizing and classifying visuals is fundamental in computer vision. It has a wide range of applications, including robotics, autonomous vehicles, intelligent video surveillance, and augmented reality. Formally, given an input image, visual object detection models generate bounding boxes tight to objects, as well as the object labels for each box (see Fig. 4). Video action detection is similar, but the bounding boxes are replaced with the starting and ending time in the temporal domain. To learn object/action detection models, fully supervised methods require instance-level annotations. Since training requires a large amount of data, it is usually a challenge to gather training data even with the help of crowdsourced platforms. Weakly supervised methods only use image-level (or video-level) labels. They look for a set of tight bounding boxes (or time intervals), such that the classification likelihood is maximized. To some extent, the weakly supervised methods alleviate the labor-intensive annotation work,



[0.515] I should buy Max Factor makeup because it is what celebrities use

[0.729] I should support women's rights because it is the right thing to do

[0.783] I should buy Lancome perfume because it will make me feel happy

[0.801] I should get the most affordable insurance because esurance could save me money

[0.835] I should wear Dior because Sharon Stone wears it

[0.348] I should drink Bud Light because I have good taste

[0.384] I should drink this beer because it tastes good

[0.430] I should drink Pilsner Beer because it will associate me with standards

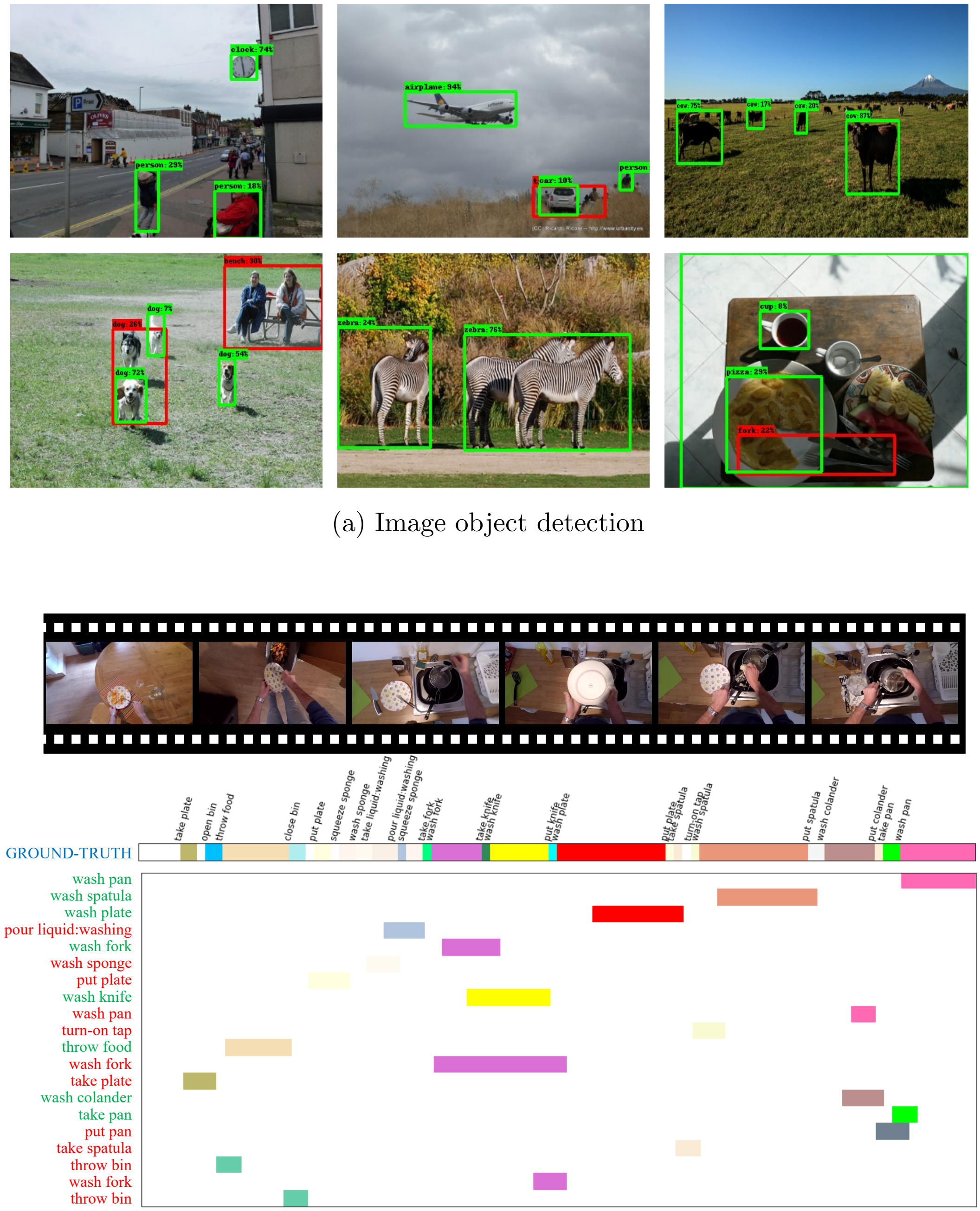
[0.440] I should drink Schaefer beer because it is a fine tasting beer

[0.462] I should buy coors light because it is the worlds most refreshing can

Figure 3: Ranking examples of the ads understanding task. Systems have to rank the paired action-reason statement lower. We show the paired statement in **bold** and the predicted image-text distance in the brackets.

yet they still need a crowdsource environment to provide clean and non-noisy labels.

In the thesis, Chapter 7, 8, 9 target the same goal of harvesting detection models to recognize and localize visual instances. For example, Fig. 4a shows the results from Chapter 7, which generates detection boxes (green and red) for an given image. Fig. 4b shows some predictions from Chapter 9, which detects time intervals (starting and ending time) as well as the action labels associated. The common challenge for them is similar to visual reasoning, i.e., how to deal with noise in natural supervisions and how to extract reliable information for localization. We provide answers in these chapters.



(a) Image object detection

(b) Video action detection

Figure 4: Visual detection examples. The localization for both image objects and video actions can be evaluated using the Average Precision at a given overlap/IoU threshold. We show correct (overlap/IoU > 50%) detections in green, incorrect ones in red.

1.5 Contributions

We summarize the contributions of this thesis. First, we show how to use multimodal features as well as the encoded external knowledge to understand advertisements better. Although we verify using our ads dataset, the method, in theory, can be generalized to other data that requires to understand different types of rhetorics (e.g., story or movie understanding). Moreover, the use of external knowledge also fits the VQA and Image-Text retrieval background, it hence could be easily adapted.

Second, we have studied two cases (one uses our ads dataset [94] while the other uses the VCR [314]) showing that models utilize shortcuts instead of real evidence to make decisions. These studies remind researchers to take care of the shortcut effects to fully unleash the power of multimodal features. Besides the observational studies, we also provide robust training solutions to overcome the negative effects, resulting in generalizable models.

Last but not least, we explore the new direction of injecting the multimodal signals into the supervisions. On one hand, we propose a task to learn object detection models from weak supervision of textual captions. It is more useful than weakly supervised object detection (WSOD) using ideal image-level labels since it suits a more practical using scenario. In such a use scenario, tons of web images paired with descriptive captions or user-generated tags can be potentially used for training the models. On the other hand, we attempt to learn video action detection models from audio narrations. We find the audio narrations to be both cheap and informative, leading to good action localization performance.

2.0 Related Work

In this chapter, we retrospect the related techniques for our two primary focus: multimodal knowledge integration for visual reasoning (Sec. 2.1) and multimodal cues for localizing objects and actions (Sec. 2.2).

2.1 Visual Reasoning with Multimodal Features and External Knowledge

Visual reasoning is a challenging but important task that is gaining momentum. Example tasks include reasoning about what will happen next in film, or interpreting what actions an image advertisement prompts. Both of these reasoning tasks are "puzzles" which engage the viewer and invite them to combine knowledge from prior experience, in order to find the answer. Since one of the primary goals of this thesis is to study the multimodal knowledge integration for visual reasoning, we summarize in Sec. 2.1.1 the extensive multimodal features. We are also interested in using multimodal features efficiently without incorporating model biases, so we summarize in Sec. 2.1.2 the methods for evaluating, diagnosing, and overcoming model biases. Besides, we summarize the reasoning tasks such as understanding the ads and multimedia in Sec. 2.1.3.

2.1.1 Extensive Multimodal Features

Region Proposals Estimating Instance-level Attention Prior. Region proposals [81, 211, 150, 70] guide a model to regions likely to contain objects, thus for better reasoning. Attention [33, 302, 285, 228, 291, 210, 184, 153, 61, 323, 198] focuses prediction tasks on regions likely to be relevant. We show that for our task, the attended-to regions must be those likely to be visual anchors for symbolic references.

Vision, Language and Image-Text Embeddings. Recently there is great interest in joint vision-language tasks, e.g. captioning [265, 115, 52, 107, 11, 302, 262, 260, 309, 290, 64, 198, 227, 35, 125], visual question answering [12, 307, 161, 291, 228, 283, 249, 327, 328, 88, 271, 106, 251], and cross-domain retrieval [31, 27, 310, 141]. These often rely on learned image-text embeddings. [56, 121, 80] use triplet loss where an image and its corresponding human-provided caption should be closer in the space than pairs that do not match. [53] propose a bi-directional network to maximize correlation between matching images and text, akin to CCA [87]. None of these consider images with implicit persuasive intent, as we do. We compare against [56, 53] in Sec. 3.3.

External Knowledge for Vision-language Tasks. [283, 271, 106, 328, 251] examine the use of knowledge bases and perform explicit reasoning for answering visual questions. [262] use external sources to diversify their image captioning model. [174] learn to compose object classifiers by relating semantic and visual similarity. [164, 73] use knowledge graphs or hierarchies to aid in object recognition. These works all use mappings that are objectively/scientifically grounded, i.e. lion is a type of cat. In contrast, we use cultural associations that arose in the media/literature and are internalized by humans, e.g. motorcycles are associated with adventure.

Human Emotions. Human emotions are also an important cue for visual reasoning. Researchers have been interested in predicting facial expressions and emotions for a long time [54, 111, 42]. Large datasets exist [178, 20, 122]. We train a facial expression model on [178] and apply it on faces detected in the video, as a cue for the viewers' sentiment.

Video Dynamics and Actions - Motion, Pose, and Activity. Optical flow [58, 24, 241, 166, 96, 207] is a basic building block of video understanding. We use [207] due to its simplicity and reliable accuracy. Higher-level analysis of video includes human pose estimation [255, 229, 188] and action detection and recognition [300, 71, 268, 28]. Unlike these, optical flow does not capture semantics (such as the name of the action performed in a video). This is desirable in our case since a wide variety of activities can be exciting and climactic, so categorization is less useful. Anomaly detection [159] is also related, but rather than predicting what does not fit, we wish to predict how a video builds up and increases its dramatic content to create the climax.

2.1.2 Model Biases: Evaluation, Diagnosis, and Overcoming Them

Formulation and Evaluation of Reasoning Tasks. A widely accepted definition for reasoning involves utilization of external information (from a knowledge base or pretraining on multimodal data in unsupervised fashion), which helps to answer questions about an image. Visual question answering (VQA) is a representative task. It asks vision-related multichoice questions and assumes that only the models armed with reasoning capabilities could answer. Early benchmarks [12, 77, 319] provide only the image and paired question/answers, but various approaches [251, 283] have explored incorporating diverse external resources. Later on, knowledge-based VQA (KBVQA) datasets such as [163, 225, 232, 271, 270] provide facts or background knowledge as part of the dataset release. They facilitate the adoption of external knowledge in VQA algorithms: for example, [185, 186] predict whether the answer is in the knowledge base and further choose the most suitable answer from candidates. The weakness of both VQA and KBVQA, we argue, is they over-simplify the reasoning by asking a *single* question, i.e. while answering is explicitly evaluated, reasoning evaluation is only implicit. This setting is not suitable for verifying the effectiveness of external knowledge usage. Many works studied the VQA/KBVQA benchmark validity, e.g. [77, 319] retrospected on organizing the VQA challenge and proposed methods to improve the datasets. [205] studied the language priors in the VQA dataset, and forced the method to look at the image; we instead force it to look at external knowledge. In the NLP domain, [101, 104, 272] showed on the SQuAD [204] benchmark that providing *always-relevant* knowledge is not a good practice since the learned models are not necessarily based on the facts to reason. They turned questions into confusing facts and added them to the knowledge context. They observed state-of-the-art models to be fragile to such a simple input change.

To alleviate the oversimplification issue in VQA, the visual commonsense reasoning (VCR) task [314] requires models to both answer a visual question, and provide rationales to justify the answers. Automatic understanding of advertisements [94] requires the comprehension of two aspects of the ads: one requires the machine to answer what action is suggested in the ad, and the other requires the model to explain the ad's arguments for encouraging this action. However, both the VCR and Ads understanding tasks fail to apply hard constraints to enforce the answer prediction to be based on the rationale. In other words, the reason predictions are merely the outputs of some parallel models, which are not necessarily helpful to the primary objective.

In this work, we specifically focus on evaluating the ability of knowledge-based visual reasoning methods [163, 185, 186, 271] to retrieve relevant knowledge. In addition to the main metric (which measures the accuracy of answer prediction), we explicitly evaluate its reasoning capability, i.e., whether the model could find the correct knowledge piece to use. Our new side task detected models that are utilizing superficial patterns.

Dataset Biases and Diagnosis. Many works studied VQA dataset biases to improve data acquisition. For example, [77, 319] optimized the annotation pipeline to cope with questions being answerable without examining the visual contents. The problem we study is orthogonal as it has to do with question-answer shortcuts rather than the presence of modes in the answer class distribution. The VCR authors [314] trained an adversarial matching model to provide suggestions for the distracting options, but we show shortcuts still exist. [91] developed a question engine to leverage scene graph structures to dispatch diverse reasoning *questions*, thus tightly control the answer distribution; this does not remove question-answer shortcuts. [105] proposed the procedurally generated *synthetic* CLEVR dataset and minimized the biases of the annotations through random sample generation; this is not possible for VCR. [3] propose train-test splits that have different answer distribution priors, but over-reliance on priors is not the only problem. Importantly, prior work has largely focused on biases in the classification probability given the question, but the shortcuts we study take the broader form of co-existing words or objects, in the question-answer pair. Methods to cope with classification biases do not apply also because both question and answer are entire sentences.

Constructing adversarial data to attack the trained models is a way to diagnose the effects of dataset biases, and we propose a technique in our work. In *text* question answering (QA), [101, 272, 104] applied adversarial evaluation on the SQuAD dataset [204]. They turned questions into confusing facts that should have no impact on the answers and added them to the knowledge context to distract models. Our strategy for modifying the evaluation is (1) simpler, i.e. we only replace pronouns with existing person tags or we mask, rather than generating new content at the phrase level, and (2) we are not aware of prior adversarial evaluation in the VCR setting.

Explainable Models. Our focus is on ensuring and evaluating a model's ability to select reliable evidence (i.e. external knowledge), *not* on the explainability/interpretability of models to a human. In other words, we care about the correctness of knowledge pieces used, rather than how interpretable the model's selections are. Because the difference is subtle, we briefly discuss explainable models and how they are different than our work. Some work [83, 93] collects explanation annotations and requires a model to point to the human-annotated reasons for an effect—for example, finding the spatial location in an image that directly affects a model's prediction. Unlike our work, these require annotation effort, i.e. humans provide explanations. They also resemble the parallel reasoning task as evidence does not necessarily lead to the main model decision. Attention mechanisms and Graph Convolutional Networks (GCN) [120] are another way to achieve explainability. They optimize a primary goal, meanwhile, learn the reliability of different evidence. Approaches such as [153, 184, 185, 198, 220, 228, 285, 286, 291, 302] fall into this category. Our approach is similar in that we do not require additional supervision, but we propose a new side task to explicitly evaluate the model's ability to choose the right evidence. We study the relation between *choosing correct supportive evidence* and *predicting the correct answer*.

Methods that Lead to Robust Training. General-purpose techniques, e.g. dropout [238], regularization, or pre-training, potentially benefit VL tasks. In NLP, distributed representations [173, 200] are often used to initialize sequence models. ELMo [201] and BERT [50] learn context word embeddings through left-to-right/right-to-left or masked language modeling, and are often used for pretraining in downstream tasks. In vision-and-language, [37, 152, 140, 240] extend BERT to the multi-modal setting to pre-train on large VL datasets e.g. Conceptual Captions [226]. The methods we study also use various forms of pre-training, but still suffer from shortcut effects.

To cope with specific dataset biases, [205] push their full VQA model away from a question-only one, thus encourage the former to pay more attention to the visual features. [156] train textual distractors using reinforcement learning to confuse the answering module thus partially resolve the priors in question type. However, these are limited to the classification setting, and tackle a different type of bias. We instead propose a technique to cope with the input-output shallow matches.

Alternative Methods for VQA. There are methods that aim to perform "true reasoning" as we have described in it Sec. 6.1, e.g. neural module networks, executable programs, and neuro-symbolic methods [9, 106, 301, 162, 261] which break up reasoning into a sequence of steps. However, these are predominantly applied on synthetic VQA settings (e.g. CLEVR) and are not appropriate for the VCR dataset.

2.1.3 Automatic Understanding of Advertisements and Multimedia

Predicting Placement and Responses to Ads. We are not aware of any work in decoding the meaning of advertisements as we propose. However, [16, 39] predict click-through rates in ads using low-level vision features, whereas we predict what the ad is about and what message it carries. [288, 168] determine the best placement of a commercial in a video stream, or of image ads in a part of an image using user affect and saliency. [216, 67] detect whether the current video shown on TV is a commercial or not, and [224] detect human trafficking advertisements. [256] modify ads to be indiscernible from regular images, in order to bypass ad-blockers. In terms of human responses to ads, [167] predict how much human viewers will like an ad by capturing their facial expressions. Human facial reactions, and ad placement and recognition, are quite distinct from our goal of decoding the messages of ads. There is also extensive research in the media studies, communications and advertising research community [278, 170, 222] on how ads build rapport, but this research is not computational.

Predicting Effectiveness. Media arts papers have examined effectiveness as related to context [239], repetition [234], brand recognition [139], emotion and engagement [250], intellectual curiosity [235], as well as humor, iconic characters, and thought-provoking content [59]. Most of these papers require human input from surveys, and do not perform computational analysis or automatic prediction of effectiveness. [127] considers specific demographics of individual viewers, as well as the final result of a video's views to determine effectiveness, which precludes inferring quality before a video is released. Finally, [206] uses neural networks to predict TV ad effectiveness, but all 837 participants in the survey analyzed one of three ads, all of which were marketing toothpaste. Our dataset consists of close to forty topics and product types, thus making the task more challenging.

Retrieving the Most Suitable Ads-related Statements. The early work [94] proposes the problem of decoding ads, formulated as answering the question "*Why* should I [action]?" where [action] is what the ad suggests the viewer should do, e.g. buy a car or help prevent domestic violence. The dataset contains 64,832 image ads. Annotations include the topic (product or subject) of the ad, sentiments and actions the ad prompts, rationales provided for why the action should be done, symbolic mappings (referred to as signifier-signified, e.g. motorcycle-adventure), etc.

In [94], ads understanding was defined as a classification task. [294] proposed a cross-modal retrieval task to match the action-reason statement provided by human annotators ("What action should the viewer take based on the ad? What reason does the ad provide for taking the suggested action?"). [297, 4, 195, 294] proposed models for the cross-modal retrieval task, where [294] incorporated knowledge from captioning and symbol prediction models, [4] used a symbolism-based attention model, and [195, 297] additionally used textual slogans in the image extracted with OCR techniques. Instead of using an embedding from a single modality or fusing the multi-modal features, [298] used a graph and allow message passing between modalities. The learned weights in the graph structure capture the model's reasoning and can be used to gauge "How does the model incorporate external knowledge to reason about an ad?".

Automated Media Analysis There is a small body of work in analyzing persuasion and social phenomena as portrayed in the media domain. [109] analyze in what light a photograph portrays a politician, and [110, 273] examine how the facial features of a candidate determine the outcome of an election. [199] examine the facial attributes of faces in politics, and [252] examine the variance of faces in ads. Some work also analyzes events (e.g. protests) as reported in social media [280]. This work primarily applies to images of people. Also related is work in parsing infographics, charts and comics [25, 117, 97]. In particular, these focus on modeling attention, or extracting information and answering questions about comic books. In contrast to these, our interest is analyzing the *implicit* arguments ads were created to make.

Movie and Story Understanding One task in our work is to understand the structure of video ad stories. Others have developed techniques for understanding movie plots [249, 182, 263] and the principal characters and their relations [277]. While there is no prior work on detecting climax in ads, some previous approaches model the tempo of other videos. [149] use cues like "motion intensity" and "audio pace" to detect action scenes. [208] use pacing to recognize movie genre since action movies are faster-paced than dramas. [40] create video stories out of consumer videos, using story composition and dynamics. We show semantic context features based on objects, scenes and emotions improve the performance of purely motion- or pace-based ones.

2.2 Multimodal Cues for Localizing Objects and Actions

Object and action detection deal with recognizing and localizing instances of specific visual categories (e.g., person, car, bus, cat, etc.). The former generates bounding boxes tight to objects and the labels for each box, while the latter predicts start and end times for specific actions. To achieve the two basic models, fully supervised object or action detection requires instance-level annotations for all objects in images or all actions in videos. However, it is labor-intensitive to get the training datasets. Thus, the two tasks are not easy to scale to use big data. Weakly supervised detection aims to alleviate the burden of collecting such expensive box/action annotations. It requires only image-/video-level supervision, thus gathers attention. However, weakly supervised detection still needs an unnatural and crowdsourced environment. It inspires us to look for an entirely free method without human annotation effort. That is the reason we explore multimodal cues for localization. We summarize methods that learn from caption supervision (Sec. 2.2.1), and briefly introduce the approaches applied to videos (Sec. 2.2.2).

2.2.1 Learning Image Object Detectors from Weak Caption Supervisions

Weakly Supervised Detection via MIL. Most Weakly Supervised Object Detection (WSOD) methods formulate the task as a multiple instance learning (MIL) problem. In this problem, proposals of an image are treated as a bag of candidate instances. If the image is labeled as containing an object, at least one of the proposals will be responsible to provide the prediction of that object. [194, 324] propose a Global Average (Max) Pooling layer to learn class activation maps. [22] propose Weakly Supervised Deep Detection Networks (WSDDN) containing classification and detection data streams, where the detection stream weighs the results of the classification predictions. [114] improve WSDDN by considering context. [248, 247] jointly train multiple refining models together with WSDDN, and show the final model benefits from the online iterative refinement. [51, 275] apply a segmentation map and [275] incorporate saliency. Finally, [266] adds a min-entropy loss to reduce the randomness of the detection results. Our work is similar to these since we have also attempted to represent the proposals using a MIL weighted representation, however, we go one step further to successfully adopt a more advanced neural architecture, and a more challenging supervision scenario.

Weakly Supervised Scene Graphs Generation. Since the captions we proposed to use also include the relation information, we learn the object detector and relation detector in a joint manner — which serve the same goal as weakly supervised scene graphs generation. Most weakly supervised scene graphs generation methods [34, 78, 141, 148, 151, 187, 203, 284, 289, 315] learn to generate graphs in a fully-supervised manner, in which training data involves both entities (bounding boxes and labels) and predicates. Inspired by weakly-supervised object detection (WSOD) [22, 194], [202, 313, 317] somewhat reduce the reliance on these *labor-intensive* annotations. [202] infer visual relations using only image-level triplets. [317] directly apply WSOD for entity localization and add a weakly-supervised visual relation detection (WSVRD) task for classifying entity pairs. [313] match predicates to entities and jointly infer the entities, predicates, and their alignments, using a bipartite graph. However, [202, 313, 317] still require clean triplet annotations from crowdsourcing, while our method only needs captions. Further, we capture visual properties in the internal graph at training time; these cannot be represented using triplets but help to enrich the visual representation and better ground entities. [313]'s method includes a more general (subject, predicate, $\emptyset$) graph, but it does not capture visual attributes.

Understanding Text and Learning Visual Objects from it. Recently there has been great interest in modeling the relationship between images and texts, but to our knowledge, no work has explored learning a detector for images from captions. [31] learn to discover and localize new objects from documentary *videos* by associating subtitles to video tracklets. They extract keywords from the subtitles using TFIDF, but we show that only using words that actually appear in the caption (as done with TFIDF) results in suboptimal performance.

There is also work to associate phrases in the caption to visually depicted objects — visual grounding of phrases. It locates the entities in an image, based on a given natural language query. [116] align sentence fragments with image regions. [32, 213] attend to the relevant image regions to reconstruct the input phrase, similar to weakly-supervised object detection. [321] incorporate a spatial transformer [98] to refine object boxes relative to multi-scale anchors. However, none enable training of an independent object detector with accurate localization and classification, as we propose.

Besides, a group of work is trying to parse the captions and use the structural parsing results to help understanding visual objects. Open information extraction systems [10, 46, 55, 165, 292] produce relation triples using surface and dependency patterns, but target language-only relation extraction or question answering. On the vision end, method exist to parse a question or image into a structured, tree-like form, for composable visual reasoning [9, 72, 106, 118, 162, 301]. Following the emergence of scene graphs [108] as a global description of an image, automatic parsing from textual descriptions to scene graphs [219, 274] aims to fill the gap between texts and images. It tackles practical issues such as pronoun resolution and plural nouns, and duplicates some nodes in the scene graph if necessary. Though we use the parser designed in [219], our reliance on parsing is different. While the above methods tackle pure language tasks, visual question answering, and image retrieval, we use the parsed results as supervised signals to guide a scene graph generation model during training. Our work is similar to [31, 100, 299] since we extract or amplify information from captions. However, these works only extract *entities* from captions, while we also learn from the properties and relations described. Also related are recent methods that use supervision from visual-language pairs [48, 171, 183, 242, 282], but these learn general-purpose representations and do not perform scene graph generation.

Captions, Categories and Human Bias. We notice that there is a gap between what humans name in captions, and what categorical annotations they provide. [175] study a similar phenomenon they refer to as "human reporting bias". They model the presence of an actual object as a latent variable, but we do the opposite—we model "what's in the image" by observing "what's worth saying". Further, we use the resultant model as precise supervision to guide detection model learning. In other work, [318] predict the nuance between an ad image and a slogan, [258] study attribute dominance, and [21] explore perceived visual importance.

2.2.2 Learning Weakly Supervised Models from Videos

Video Datasets Involving Objects and Actions. To deal with experiments on the video data, many datasets were proposed. Wildlife Documentaries Dataset [31] contains 15 documentary films from YouTube and each is 9 - 50 minutes long. The authors provide tracklet-level annotations for weakly supervised object detection evaluation. HowTo100M instructional videos [171] contains 1.2M videos with automatically generated speech transcription. Their videos feature the daily objects and actions on instructional videos. The EPIC-Kitchens dataset [45, 44] records egocentric videos regarding actions in the kitchen. They even annotated bounding boxes of essential objects, thus are potentially useful for localization evaluation.

Weakly Supervised Learning in Videos. Since videos naturally involve multiple modalities, many approaches use unsupervised or weakly supervised training to learn better video representations. For example, [15, 30, 189, 196] explore the cross-modal relations and leverage large amounts of unlabeled video for training. The basic idea behind this is that vision and sound are naturally synchronized so that models can utilize the synchronization as weak signals instead of ground truth labels. However, these methods are more often used in pre-training to improve the initialization of the visual-sound models. In comparison, our focus is to use the additional modalities (e.g., audio narrations) to localize visual objects or actions in the temporal domain.

Also related is the co-localization or audio-visual correspondence [2, 13, 14, 66, 80, 223]. Similar to learning the joint representations, these works also rely on the synchronization of different modalities. However, they further learn to localize the sounds or visual objects given the information from other modalities. Our work still differs from them in that 1) these works did not quantify their results on detection tasks while only provide qualitative results; 2) our model requires no supervised signals at testing time.

Weakly Supervised Video Detection Tasks. Extending the weakly supervised detection to the video domain is required to greatly reduce the expensive human efforts. However, due to the different understandings caused by the additional time axis, weakly supervised object detection has various definitions in videos. For example, [129, 133, 142, 190, 177, 179, 269] only learn to detect the starting and ending time of particular actions, while entirely ignored spacial layouts of the instances. To track the spatio-temporal localization, methods such as [31, 276] rely on the video/image proposal frameworks such as [113, 112, 99, 259] which provide high-quality region proposals. Their approaches are counterparts to the WSOD of the image domain, with the only difference in the types of proposals. Finally, there are also methods [176, 130, 132] attempted to only utilize cues from videos (e.g., motion, subtitle, tight boxes) to potentially benefit the training of image detectors.

We study how to learn the action detection models (predicting starting/ending time and action labels) in videos in the thesis. However, as compared to fully-supervised methods [57, 71, 157, 231, 236, 300, 322], the supervised signals we used are the audio narrations, which are noisy in nature hence are much weaker than instance-level annotations. As for the weakly supervised action detection models [129, 190, 191, 197, 230, 269], their data in most cases only involve one single action per video. Thus video-level supervision satisfy their requirements. In comparison, our targeting task is a novel and new task, requiring non-trivial efforts to deal with the noisy annotations to improve the learned model's quality.

3.0 ADVISE: Symbolism and External Knowledge for Decoding Advertisements

3.1 Introduction

Many visual reasoning tasks require to understand the multimodal cues and the content referring to the knowledge outside the image/video. Thus, one needs to know the background to understand them. E.g., a photo of a celebrity in public media may refer to his/her recent public statement regarding a social event. In this chapter, we focus on advertisements since they are well-designed to incorporate complicated and informative features from external knowledge, hence are excellent resources for testing model reasoning.

Advertisements are a powerful tool for affecting human behavior. Product ads convince us to make large purchases, e.g. for cars and home appliances, or small but recurrent purchases, e.g. for laundry detergent. Public service announcements (PSAs) encourage socially beneficial behaviors, e.g. combating domestic violence or driving safely. To stand out from the rest, ads have to be both eye-catching and memorable [303], while also conveying the information that the ad designer wants to impart. All this must be done in a limited space (one image) and time (however many seconds the viewer spends looking at the ad).

How can ads get the most "bang for their buck"? One technique is to make references to knowledge viewers already have, e.g. cultural knowledge, associations, and *symbolic mappings* humans have learned [221, 136, 237, 135]. These symbolic references might come from literature (e.g. a snake symbolizes evil or danger), movies (a motorcycle symbolizes adventure or coolness), common sense (a flexed arm symbolizes strength), or pop culture (Usain Bolt symbolizes speed).

In this chapter, we focus on the task of *inferring the suggested action* (what the viewer should do) and *provided arguments* (why they should do it, according to the ad). We propose a novel method that embeds images/videos and action-reason (what-why) statements, to allow retrieval of statements given an image/video.

We first describe how to use symbolic mappings to predict the messages of image advertisements. On one hand, we model how components of the ad image serve as visual anchors to concepts outside the image, using annotations in the Ads Dataset of [94]. On the other hand, we use knowledge sources external to the main task, such as object detection models, to better relate ad images to their corresponding messages. Both of these are forms of using outside knowledge, and they both boil down to learning links between objects and symbolic concepts. We use each type of knowledge in two ways, as a constraint or as an additive component for the learned image representation. We show that the knowledge as an additive component helps to improve the ads understanding performance, while the knowledge as a constraint additionally helps explain the symbolic links spatially in an image.

Then, we extend the multimodal features used in the basic image model and further design a model for the video ads. In comparison, this time we focus more on the multiple modalities, i.e., the contributions of image/video and text/speech/sound.

To summarize, our contributions are as follows:

- We show how to effectively use symbolism to better understand image ads.
- We show how to make use of noisy caption predictions to bridge the gap between the abstract task of predicting the message of an ad, and more accessible information such as the objects present in the image. Detected objects are mapped to symbols via a domain-specific knowledge base.
- We show how to encode video ads as a bag of frames or a sequence of frames.
- We show how embedded slogans and speech are influencing image and video ads understanding, respectively.

3.2 Approach

We focus on the following multiple-choice task, implemented via ranking: Given an image/video ad and several statements, the system must identify the correct statement to pair with the ad. For example, for test image D in Fig. 5, the system might predict the right statement is "Buy this drink because it's exciting." This ranking task is akin to multiple-choice question-answering, which was also used in prior VQA works [12, 249], but unlike

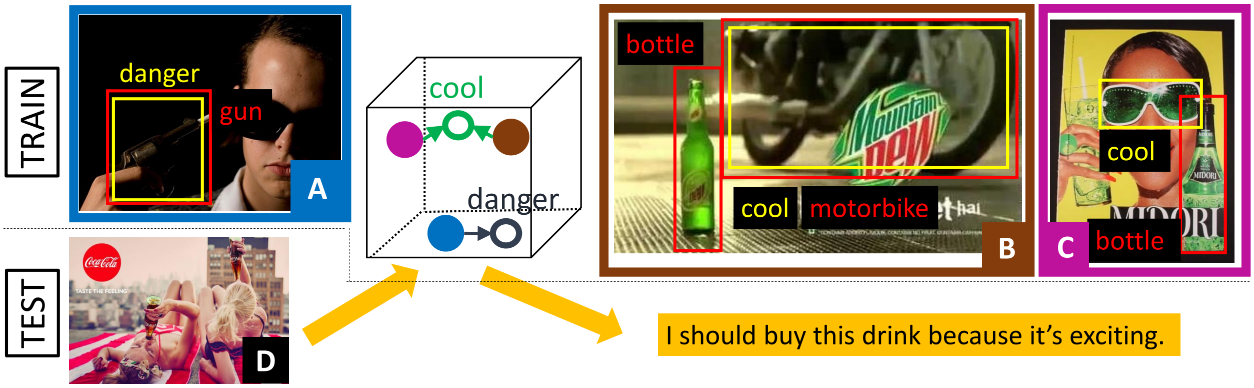


Figure 5: Our key idea: Use symbolic associations shown in yellow (a gun symbolizes danger; a motorcycle symbolizes coolness) and recognized objects shown in red, to learn an image-text space where each ad maps to the correct statement that describes the message of the ad. The symbol "cool" brings images B and C closer together in the learned space, and further from image A and its associated symbol "danger." At test time (shown in orange), we use the learned image-text space to retrieve a matching statement for test image D. At test time, the symbol labels are *not* provided.

these, we do not take the question as input. Similarly, in image captioning, [115, 56] look for the most suitable image description.

We learn an embedding space where we can evaluate the similarity between ad images (or videos) and ad messages (Sec. 3.2.1). We show how to use symbolic mappings to predict the messages of image advertisements (Sec. 3.2.2). Then, we focus on the multiple modalities. We present our image/video models, which consider multimodals from image/video and text/speech/sound (Sec. 3.2.3).

3.2.1 Cross-modal Triplet Embedding for Statements Retrieval

We first directly learn an embedding that optimizes for the retrieval/ranking task. We require that the similarity between an image (video) and its corresponding statement should be higher than the similarity between that image (video) and any other statement, or between other images (videos) and that statement. Thus, we minimize Eq. 1:

$$L(\boldsymbol{v},\boldsymbol{t};\boldsymbol{\theta}) = \sum_{i=1}^{K}\Big[\underbrace{\sum_{j\in N_{vt}(i)}\max\left(0, \frac{\boldsymbol{v}_i^\intercal\boldsymbol{t}_j}{\|\boldsymbol{v}_i\|\|\boldsymbol{t}_j\|} - \frac{\boldsymbol{v}_i^\intercal\boldsymbol{t}_i}{\|\boldsymbol{v}_i\|\|\boldsymbol{t}_i\|} + \beta\right)}_{\text{image (video) as anchor, rank statements}} + \underbrace{\sum_{j\in N_{tv}(i)}\max\left(0, \frac{\boldsymbol{t}_i^\intercal\boldsymbol{v}_j}{\|\boldsymbol{t}_i\|\|\boldsymbol{v}_j\|} - \frac{\boldsymbol{t}_i^\intercal\boldsymbol{v}_i}{\|\boldsymbol{t}_i\|\|\boldsymbol{v}_i\|} + \beta\right)}_{\text{statement as anchor, rank images (videos)}}\Big] \tag{1}$$

where K is the batch size; β is the margin of triplet loss; $\boldsymbol{v}$ and $\boldsymbol{t}$ ($\boldsymbol{v},\boldsymbol{t} \in \mathbb{R}^{200\times 1}$) are the visual and textual embeddings we are learning, respectively; $\boldsymbol{v}_i$, $\boldsymbol{t}_i$ correspond to the same ad and $\frac{\boldsymbol{v}_i^\intercal\boldsymbol{t}_i}{\|\boldsymbol{v}_i\|\|\boldsymbol{t}_i\|}$ measures the *cosine similarity* between the paired visual and textual embeddings; $N_{vt}(i)$ is the negative statement set for the i-th image (video), and $N_{tv}(i)$ is the negative visual set for the i-th statement, defined in Eq. 2. These two negative sets involve the most challenging k' examples within the size-K batch. A natural explanation is that Eq. 2 seeks to find a subset $A \subseteq \{1, ..., K\}$ which involves the k' most confusing examples.

$$N_{vt}(i) = \underset{A\subseteq\{1,...,K\},|A|=k'}{\arg\max}\sum_{j\in A, i\neq j}\frac{\boldsymbol{v}_i^\intercal\boldsymbol{t}_j}{\|\boldsymbol{v}_i\|\|\boldsymbol{t}_j\|}, \qquad N_{tv}(i) = \underset{A\subseteq\{1,...,K\},|A|=k'}{\arg\max}\sum_{j\in A, i\neq j}\frac{\boldsymbol{t}_i^\intercal\boldsymbol{v}_j}{\|\boldsymbol{t}_i\|\|\boldsymbol{v}_j\|} \tag{2}$$

Hard Negative Mining. Different ads might convey similar arguments, so the sampled negative may be a viable positive. For example, for a car ad with associated statement "I should buy the car because it's fast", a hard negative "I should drive the car because of its speed" (provided on another image) may also be proper. Using the k' most challenging examples in the size-K batch (Eq. 2) is our trade-off between using all and using only the most challenging example, inspired by [56, 218, 281].

Text Embedding. We use either mean-pooling or an LSTM model [84] depending on our different needs, to encode the word embedding vectors (initialized from GloVe [200]) into 200-D statement embedding $\boldsymbol{t}$. For Sec. 3.2.2, we focus on the symbolic mappings, thus require the permutation-invariant nature of mean-pooling. For Sec. 3.2.3, we use LSTM for the image ads understanding and mean-pooling for the video understanding model, considering the performance.

Basic Image Embedding. As a very basic representation, we extract the Inception-v4 feature [243] of a full image $\boldsymbol{x}$, denoted as $\phi_{cnn}(\boldsymbol{x}) \in \mathbb{R}^{1536\times 1}$. Then we use a fully-connected layer with parameter $\boldsymbol{w}_{img} \in \mathbb{R}^{1536\times 200}$ to project it to the joint embedding space, resulting in $\boldsymbol{v} = \boldsymbol{w}_{img}^{\intercal}\phi_{cnn}(\boldsymbol{x}) \in \mathbb{R}^{200\times 1}$.

Basic Video Embedding. For the video ads, we treat the video as a *bag of frames* (BOF) which ignores the sequence order, due to the limited size of our video ads dataset. Consider a frame sequence $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_R$ of a video where R is the total number of frames. Given that we sample 1 frame per second, R also equals to the time duration measured in seconds. We use Inception-v4 [243] to extract frame features, resulting in $\phi_{cnn}(\boldsymbol{x}_1), \ldots, \phi_{cnn}(\boldsymbol{x}_R)$. The BOF model then uses mean-pooling to represent the video $\boldsymbol{z}_{vdo} = \frac{1}{R}\sum_{i=1}^{R} \boldsymbol{w}_{vdo}^{\intercal}\phi_{cnn}(\boldsymbol{x}_i)$, where $\boldsymbol{w}_{vdo} \in \mathbb{R}^{1536\times 200}$ is the parameter.

3.2.2 Symbolism and External Knowledge for Decoding Image Ads

We consider using the symbolic mappings to predict the messages of image ads. In addition to Sec. 3.2.1, our method involves three components: (1) an image embedding which takes into account individual regions in the image, (2) constraints on the learned space from symbol labels and object predictions, and (3) an additive expansion of the image representation using a symbol distribution. These three components are shown in Fig. 6, and all of them rely on external knowledge in the form of symbols and object predictions. Note that we can recognize the symbolic association to danger in Fig. 5 via two channels: either a direct classifier that learns to link certain visuals to the "danger" concept, or learning associations between actual *objects* in the image which can be recognized by object detection methods (e.g. "gun"), and symbolic concepts.

Image Embeddings using Symbol Regions. Since ads are carefully designed, they may involve complex narratives with several distinct components, i.e. several regions in the ad might need to be interpreted individually first to decode the full ad's meaning. Thus, we represent an image as a collection of its constituent regions, using an attention module to aggregate all the representations from different regions.

Importantly, the chosen regions should be those likely to serve as visual anchors for symbolic references (such as the motorcycle or shades in Fig.5, rather than the bottles). Thus we consider all the 13,938 images, which are annotated as containing symbols, each

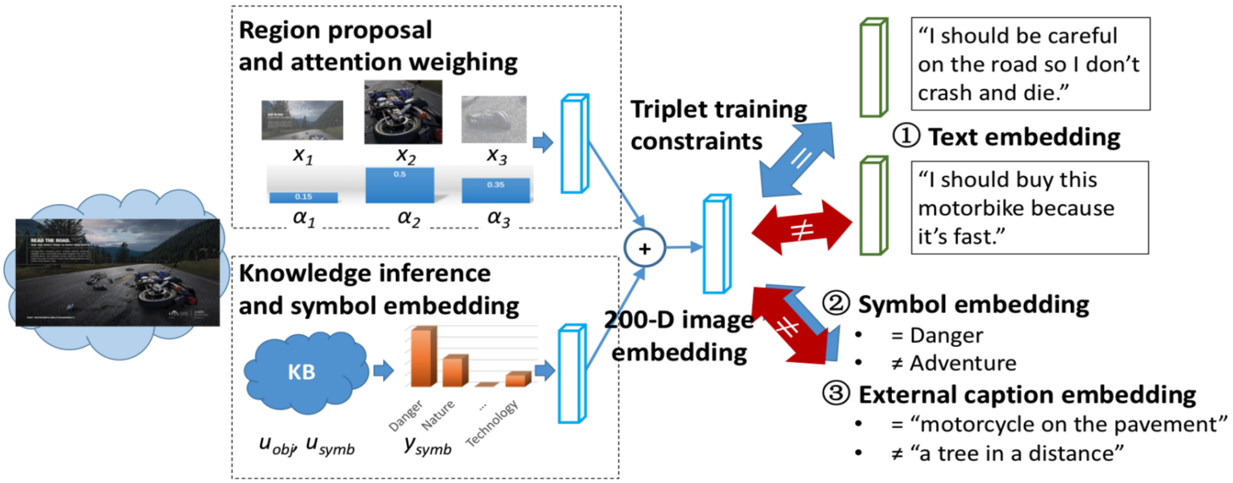


Figure 6: Our image embedding model with knowledge branch. In the main branch (top left), multiple image symbolic anchors are proposed. Attention weighting is applied, and the image is represented as a weighted combination of the regions. The knowledge branch (bottom left) predicts the existence of symbols, maps these to 200-D, and adds them to the image embedding. We then perform triplet training to learn such an embedding space that keeps images close to their matching action-reason statements.

with up to five bounding box annotations. Our intuition is that ads draw the viewer's attention in a particular way, and the symbol bounding boxes, without symbol labels, can be used to approximate this. More specifically, we use the SSD object detection model [150] implemented by [89], pre-train it on the COCO [147] dataset, and fine-tune it with the symbol bounding box annotations [94].

We use bottom-up attention [7, 251, 124] to aggregate the information from symbolic regions (see Fig. 6). Specifically, we use the Inception-v4 model [243] to extract CNN features for all symbol proposals $\{\boldsymbol{x}_1, \dots, \boldsymbol{x}_M\}$ (we set $M = 10$, i.e., 10 proposals per image), resulting in $\{\phi_{cnn}(\boldsymbol{x}_1), \dots, \phi_{cnn}(\boldsymbol{x}_M)\}$ where $\phi_{cnn}(\boldsymbol{x}_i) \in \mathbb{R}^{1536\times 1}$. Then, for each CNN feature $\phi_{cnn}(\boldsymbol{x}_i)$, fully-connected layers are applied to project it to: (1) a 200-D embedding vector $\boldsymbol{v}_i$ (Eq. 3, $\boldsymbol{w}_{img} \in \mathbb{R}^{1536\times 200}$), and (2) an importance score α_i (Eq. 4, $\boldsymbol{w}_{attn} \in \mathbb{R}^{1536\times 1}$). The final image representation $\boldsymbol{z}$ is a weighted sum of these region-based vectors (Eq. 5).

$$\boldsymbol{v}_i = \boldsymbol{w}_{img}^{\intercal}\phi_{cnn}(\boldsymbol{x}_i) \tag{3}$$

$$\alpha_i = \frac{\exp(\boldsymbol{w}_{attn}^{\intercal}\phi_{cnn}(\boldsymbol{x}_i))}{\sum_{j=1}^{M}\exp(\boldsymbol{w}_{attn}^{\intercal}\phi_{cnn}(\boldsymbol{x}_j))} \tag{4}$$

$$\boldsymbol{z} = \sum\nolimits_{i=1}^{M}\alpha_i\boldsymbol{v}_i \tag{5}$$

The loss to learn the image-text embedding is the same as Eq. 1, but now defined using the region-based image representation $\boldsymbol{z}$ (Eq. 5) instead of $\boldsymbol{v}$: $L(\boldsymbol{z},\boldsymbol{t};\boldsymbol{\theta})$.

We show in Sec. 3.3.1 that (1) learning a region proposal network with attention, and (2) learning from symbol bounding boxes, greatly help the statement retrieval task. In particular, statement ranking results are worse if we use a generic pre-trained region proposal network. We argue general-purpose object detection models cannot capture nuance in ads since they ignore uncommon or abstract objects.

Constraints via Symbols and Captions. We next exploit the symbol labels which are part of [94]. Symbols are abstract words such as "freedom" and "happiness" that provide additional information humans sense from the ads. We add additional constraints to the loss terms such that two images/statements that were annotated with the same symbol are closer in the learned space than images/statements annotated with different symbols. In the *extra* loss term (Eq. 6), $\boldsymbol{s}$ is the 200-D embedding of a symbol word; $\boldsymbol{z}$ is the 200-D region-based image representation defined in Eq. 5; and $N_{sz}(i)$ and $N_{st}(i)$ are the negative image/statement sets of the i-th symbol in the batch, defined similar to Eq. 2.

$$L_{sym}(\boldsymbol{s},\boldsymbol{z},\boldsymbol{t};\boldsymbol{\theta}) = \\ \sum_{i=1}^{K}\Big[\underbrace{\sum_{j\in N_{sz}(i)}\max\left(0,\frac{\boldsymbol{s}_i^{\intercal}\boldsymbol{z}_j}{\|\boldsymbol{s}_i\|\|\boldsymbol{z}_j\|}-\frac{\boldsymbol{s}_i^{\intercal}\boldsymbol{z}_i}{\|\boldsymbol{s}_i\|\|\boldsymbol{z}_i\|}+\beta\right)}_{\text{symbol as anchor, rank images}}+\underbrace{\sum_{j\in N_{st}(i)}\max\left(0,\frac{\boldsymbol{s}_i^{\intercal}\boldsymbol{t}_j}{\|\boldsymbol{s}_i\|\|\boldsymbol{t}_j\|}-\frac{\boldsymbol{s}_i^{\intercal}\boldsymbol{t}_i}{\|\boldsymbol{s}_i\|\|\boldsymbol{t}_i\|}+\beta\right)}_{\text{symbol as anchor, rank statements}}\Big] \tag{6}$$

Much like symbols, the objects found in an image are quite telling of the message of the ad. For example, environment ads often feature animals, safe driving ads feature cars, beauty ads feature faces, drink ads feature bottles, etc. However, since the Ads Dataset contains insufficient data to properly model object categories, we use DenseCap [107] to bridge the objects defined in Visual Genome [126] to the ads reasoning statements. More specifically, we use the DenseCap model to generate image captions and treat these as pre-fetched knowledge. For example, the caption "woman wearing a black dress" provides extra information about the objects in the image: "woman" and "black dress". We create additional constraints: If two images/statements have similar DenseCap predicted captions, they should be closer than images/statements with different captions. The *extra* loss term is defined similar to Eq. 6 using $\boldsymbol{c}$ for the caption representations: $L_{obj}(\boldsymbol{c}, \boldsymbol{z}, \boldsymbol{t}; \boldsymbol{\theta})$.

In our setting, word embedding weights are not shared among the three vocabularies (ads statement, symbols, and DenseCap predictions). Our consideration is that the meaning of the same surface words may vary in these domains thus they need to have different embeddings. We weigh the symbol-based and object-based constraints by 0.1 since they in isolation do not tell the full story of the ad. We found that it is not sufficient to use *any* type of label as constraint in the domain of interest: using symbols as constraints gives greater benefit than the topic (product) labels in [94]'s dataset, and this point is not discussed in the general proxy learning literature [180].

Additive External Knowledge. We describe how to make use of external knowledge that is adaptively added, to compensate for inadequacies of the image embedding. This external knowledge can take the form of a mapping between physical objects and implicit concepts, or a classifier mapping pixels to concepts. Given a challenging ad, a human might look for visual cues and check if they remind him/her of concepts (e.g. "danger", "beauty", "nature") seen in other ads. Our model interprets ads in the same way: based on an external knowledge base, it *infers* the abstract symbols. In contrast to the constraints via symbols and captions, which use the *annotated* symbols at training time, here we use a *predicted* symbol distribution at both training and test time as a secondary image representation. Fig. 6 (bottom-left) shows the general idea of the external knowledge branch. Note our model only uses external knowledge to compensate its own lack of knowledge (since we train the knowledge branch after the convergence of the visual semantic embedding branch), and it assigns small weights for uninformative knowledge.

We propose two ways to additively expand the image representation with external knowledge, and describe *two ways of setting* $\boldsymbol{y}_{symb}$ in Eq. 7. Both ways are a form of knowledge base (KB) mapping physical evidence to concepts.

KB Symbols. The first way is to directly train classifiers to link certain visuals to symbolic concepts. We learn a multilabel classifier $\boldsymbol{u}_{symb}$ to obtain a symbol distribution $\boldsymbol{y}_{symb} = sigmoid(\boldsymbol{u}_{symb} \cdot \boldsymbol{x})$. We then learn a weight α_j^{symb} for each of $j \in \{1, \ldots, C = 53\}$ symbols from the Ads Dataset, denoting whether a particular symbol is helpful for the statement matching task.

KB Objects. The second method is to learn associations between surface words for detected objects and abstract concepts. For example, what type of ad might I see a "car" in? What about a "rock" or "animal"? We first construct a knowledge base associating object words to symbol words. We compute the similarity in the learned image-text embedding space between symbol words and DenseCap words, then create a mapping rule ("[object] implies [symbol]") for each symbol and its five most similar DenseCap words. This results in a $53 \times V$ matrix $\boldsymbol{u}_{obj}$, where V is the size of DenseCap's vocabulary. Each row contains five entries of 1 denoting the mapping rule, and $V - 5$ entries of 0. Examples of learned mappings are shown in Table 5. For a given image, we use [107] to predict the three most probable words in the DenseCap vocabulary, and put the results in a multi-hot $\boldsymbol{y}_{obj} \in \mathbb{R}^{V \times 1}$ vector. We then matrix-multiply to accumulate evidence for the presence of all symbols using the detected objects: $\boldsymbol{y}_{symb} = \boldsymbol{u}_{obj} \cdot \boldsymbol{y}_{obj}$. We associate a weight α_{jl}^{symb} with each rule in the KB.

For both methods, we first use the attention weights α^{symb} as a mask, then project the 53-D symbol distribution $\boldsymbol{y}_{symb}$ into 200-D, and add it to the image embedding. This additive branch is most helpful when the information it contains is not already contained in the main image embedding branch. We found this happens when the discovered symbols are rare.

Our Final Model using Symbolic mappings. To train our final model, we use the **AD**s **VI**sual **S**emantic **E**mbedding loss to combine the L, L_{sym}, and L_{obj}:

$$\begin{aligned} L_{final}(\boldsymbol{z}, \boldsymbol{t}, \boldsymbol{s}, \boldsymbol{c}; \boldsymbol{\theta}) &= L(\boldsymbol{z} + \boldsymbol{y}_{symb}, \boldsymbol{t}; \boldsymbol{\theta}) \\ &\quad + 0.1\ L_{sym}(\boldsymbol{s}, \boldsymbol{z} + \boldsymbol{y}_{symb}, \boldsymbol{t}; \boldsymbol{\theta}) + 0.1\ L_{obj}(\boldsymbol{c}, \boldsymbol{z} + \boldsymbol{y}_{symb}, \boldsymbol{t}; \boldsymbol{\theta}) \end{aligned} \tag{7}$$

3.2.3 Interpreting the Rhetoric using Multimodal Features

Next, we consider more additive multimodal features including text slogans and knowledge from symbols and visual objects (see Fig. 7) for the image ads while we consider no external constraints (i.e., L_{sym} and L_{obj}). We also consider the speech-to-text in the video ads understanding model.

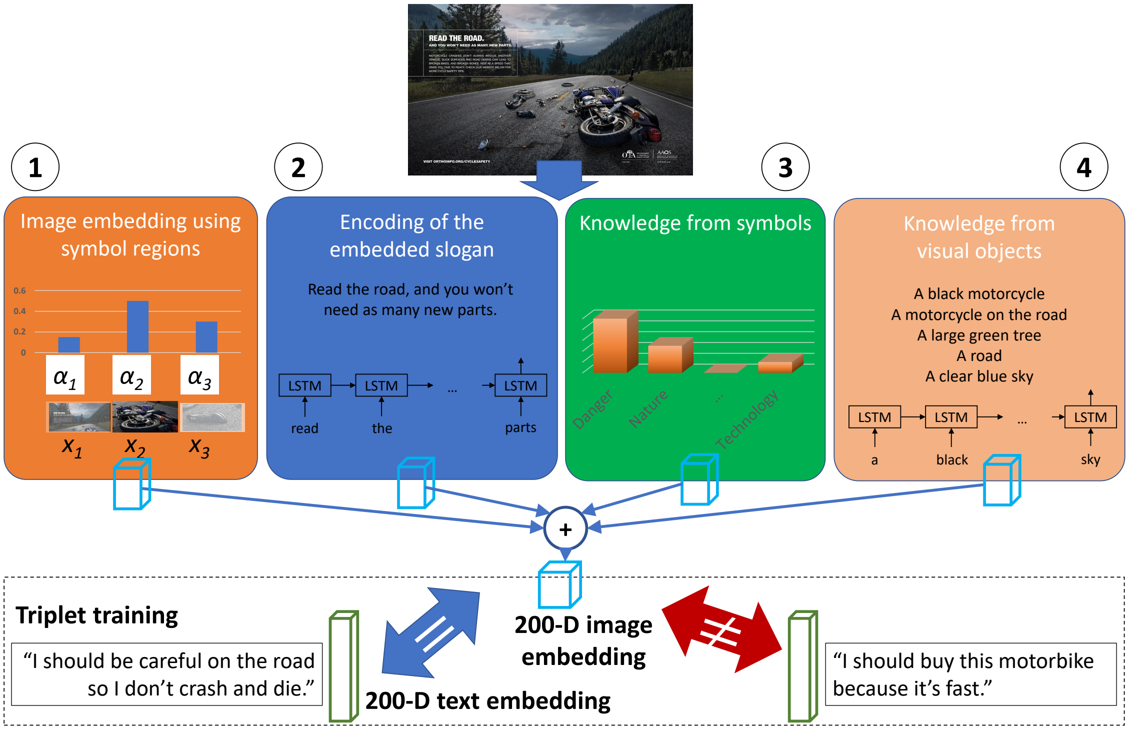


Figure 7: Our multimodal image embedding model. In the image branch (1), multiple image symbolic anchors are proposed. Attention weighting is applied, and the image is represented as a weighted combination of the regions. The knowledge branch (3) predicts the existence of symbols and maps these to the 200-D embedding. For both the slogan (2) and visual objects captions (4) branches, we use LSTM to model the phrases. Pointwise addition is applied to fuse the features from four different modalities. We then perform triplet training to learn such an embedding space that keeps images close to their matching action-reason statements.

Embedded Slogans for Understanding Image Ads. In most cases, the image alone does not tell the full story of the ad, and may be (intentionally) ambiguous [318]. Thus, we also need to consider the *slogan* embedded in the ad to accurately retrieve statements about it. Two examples are shown in Fig. 8. It is clear that the ad understanding task becomes easier if we can read the slogans in both images, namely "words can be deadly." and "Human Trafficking. Don't ignore it". Inspired by these examples, we design a method to read the slogan information and thus improve the performance of inferring actions and reasons.

Given an image, we first use the Optical Character Recognition (OCR) functionality of the Google Cloud Vision API[75] to extract the text in the ad. We concatenate all the detected pieces into one. About half of the ads have up to 20 detected tokens (usually a word or part of a word). One-fifth has between 20 and 50 tokens, 14% has between 50 and 100 tokens, and the rest of the ads have over 100 tokens. We use a standard LSTM model to obtain a slogan embedding which results in $\boldsymbol{z}_{slg} \in \mathbb{R}^{200\times 1}$. The model using only the slogan modality can be trained using $L(\boldsymbol{z}_{slg}, \boldsymbol{t}; \boldsymbol{\theta})$ in Eq. 1.

words can be deadly
think before you text
Cyberbullying has real and potentially lethal consequences.
Let's take a stand together for a change by visiting:
www.stopcyberbullying.org
Select
Back
cyberbullying
Some people are very attached to their work.
Human Trafficking. Don't ignore it.
18/10 : European Anti-Trafficking Day

(a) "Words can be deadly. Think before you text" QA: I should be careful what I say because words can hurt like any weapons.

(b) "Human Trafficking. Don't ignore it." QA: I should be aware of human trafficking because it is not always obvious.

Figure 8: Example slogans from the image ads dataset. Both images require reasoning which makes the task challenging even for a human. However, given the slogan text information, understanding the message of the ads becomes easier.

Symbols for Understanding Image Ads. We next exploit the symbol labels which are part of Ads Dataset [94]. Different from Sec. 3.2.2, our method of using symbol labels requires the training of a symbol classifier that generalizes beyond the annotated images.

However, to make the learning more feasible, instead of training classifiers on all symbols, we base our work on the 53 symbol clusters in [94]. We learn a multilabel classifier $\boldsymbol{u}_{symb} \in \mathbb{R}^{1536\times 53}$ to obtain a symbol distribution $\sigma(\boldsymbol{u}_{symb}^\intercal \phi_{cnn}(\boldsymbol{x}))$ given the feature $\phi_{cnn}(\boldsymbol{x}) \in \mathbb{R}^{1536\times 1}$, where σ is *sigmoid*. To use the additional knowledge (i.e., classifier $\boldsymbol{u}_{symb}$) regarding the symbols, we use a fully connected layer to project the symbol distribution to the joint embedding feature space, resulting in $\boldsymbol{z}_{symb} = \boldsymbol{w}_{symb}^\intercal \sigma(\boldsymbol{u}_{symb}^\intercal \phi_{cnn}(\boldsymbol{x})) \in \mathbb{R}^{200\times 1}$:

Visual Objects for Understanding Image Ads. We use the DenseCap [107] model to generate image captions and treat these as pre-fetched knowledge. Modeling the visual objects is similar to modeling the slogan. We concatenate all the captions generated by DenseCap into a long textual description, then use an LSTM model to encode the sequence, resulting in objects-text embedding $\boldsymbol{z}_{obj}$.

Speech-to-Text for Understanding Video Ads. There is more information than the visual frames that can help distinguish the video contents. For example, the audio may involve different styles of music and the speech may directly convey the ad messages. We focus on the speech information since we think they are better suited for the task of retrieving the action-reason statements. Given a video, we use FFMPEG[49] to extract the audio track. We then invoke the Google Cloud Speech-to-text API [74] to extract text from the audio data. After getting the text tokens, we concatenate them to a single sentence and use mean-pooling during training to aggregate individual word embeddings, resulting in $\boldsymbol{z}_{spch} \in \mathbb{R}^{200\times 1}$.

Our Final Multimodal Model. Our final multimodal model uses late fusion (we use *pointwise-add*) to combine the components. For the full model on the image ads, we optimize $L(\boldsymbol{z} + \boldsymbol{z}_{slg} + \boldsymbol{z}_{symb} + \boldsymbol{z}_{obj}, \boldsymbol{t}; \boldsymbol{\theta})$ (see Fig. 7). For video ads, we optimize $L(\boldsymbol{z}_{vdo} + \boldsymbol{z}_{spch}, \boldsymbol{t}; \boldsymbol{\theta})$.

3.3 Experiments

We evaluate to what extent our proposed method is able to match an ad to its intended message. We present the analysis of our symbolic mapping model (Sec. 3.2.2) in Sec. 3.3.1, including the comparison to the state-of-the-art, ablation study of components, and qualitative examples showing the learned symbolic mappings. Then, we show the contributions of extensive multimodal features, and the performance of video models (Sec. 3.2.3) in Sec. 3.3.2.

3.3.1 Analysis of the Symbolic Mapping Model.

Baselines and Evaluation Metrics. We compare our Sec. 3.2.2 to the following approaches from recent literature. All methods are trained on the Ads Dataset [94], using a train/val/test split of 60%/20%/20%, resulting in around 39,000 images and more than 111,000 associated statements for training.

- Hussain-Ranking adapts [94], the only prior method for decoding the message of ads. This method also uses symbol information, but in a less effective manner. The original method combines image, symbol, and question features, and trains for the 1000-way classification task. To adapt it, we pointwise-add the image features (Inception-v4 as for our method) and symbol features (distribution over 53 predicted symbols), and embed them in 200-D using Eq. 1 (using hard negative mining), setting $\boldsymbol{v}$ to the image-symbol feature. We tried four other ways (described in supp) of adapting [94] to ranking, but they performed worse.
- VSE++ [56] (follow-up to [121]) uses the same method as Sec. 3.2.1. It is representative of one major group of recent image-text embeddings using triplet-like losses [192, 160, 115, 212].
- VSE, which is like VSE++ but without hard negative mining, for a more fair comparison to the next baseline.
- 2-way Nets uses our implementation of [53] (published code only demoed the network on MNIST) and is representative of a second type of image-text embeddings using reconstruction losses [53, 90].

For each image, we use three related statements (i.e. statements provided by humans for this image) and randomly sample 47 unrelated statements (written for *other* images). The system must rank these 50 statements based on their similarity to the image. We compute two metrics: Rank, which is the averaged ranking value of the highest-ranked true matching statement (highest possible rank is 1, which means first place), and Recall@3, which denotes the number of correct statements ranked in the Top-3. We expect a good model to have low Rank and high Recall scores. We use five random splits of the dataset into train/val/test sets, and show mean results and standard error over a total of 62,468 test cases (removing statements that do not follow the template "I should [action] because [reason].").

Results on the Main Ranking Task. We show the improvement that our method produces over state of the art methods, in Table 3. We show the better of the two alternative methods from Sec. 3.2.2, namely KB-Symbols. Since public service announcements (e.g. domestic violence or anti-bullying campaigns) typically use different strategies and sentiments than product ads (e.g. ads for cars or coffee), we separately show the result for PSAs and products. We observe that our method greatly outperforms the prior relevant research. PSAs in general appear harder than product ads (see Sec. 3.1).

Table 3: Ranking result of Sec. 3.2.2. We show two methods that do not use hard negative mining, and three that do. Our method greatly outperforms three recent methods in retrieving matching statements for each ad. All methods are trained on the Ads Dataset of [94]. The best method is shown in **bold**, and the second-best in *italics*

Method	Rank (Lower ↓ is better)		Recall@3 (Higher ↑ is better)	
	PSA	Product	PSA	Product
2-way Nets	4.836 (± 0.090)	4.170 (± 0.023)	0.923 (± 0.016)	1.212 (± 0.004)
VSE	4.155 (± 0.091)	3.202 (± 0.019)	1.146 (± 0.017)	1.447 (± 0.004)
VSE++	4.139 (± 0.094)	3.110 (± 0.019)	1.197 (± 0.017)	1.510 (± 0.004)
Hussain-Ranking	*3.854* (± 0.088)	*3.093* (± 0.019)	*1.258* (± 0.017)	*1.515* (± 0.004)
ADVISE (ours)	**3.013** (± 0.075)	**2.469** (± 0.015)	**1.509** (± 0.017)	**1.725** (± 0.004)

Compared to 2-way Nets [53], VSE which does *not* use hard negative mining is stronger by a large margin (14-23% for rank, and 19-24% for recall). VSE++ produces more accurate results than both 2-way Nets and VSE, but is outperformed by Hussain-Ranking and our ADVISE. Our method is the strongest overall. It improves upon VSE++ [56] by 20-27% for rank, and 14-26% for recall. Compared to the strongest baseline, Hussain-Ranking

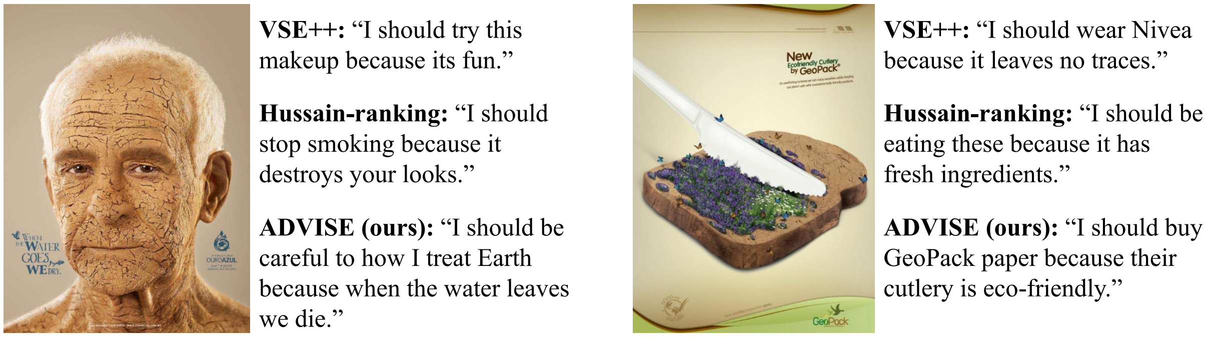
[94], our method is 20-21% stronger in terms of rank, and 13-19% stronger in recall. Fig. 9 shows a qualitative result contrasting the best methods.

Ablation Studies. We also conduct ablation studies to verify the benefit of each component of our method. We show the base triplet embedding (Eq. 1) similar to VSE++; a generic region embedding using image regions learned using [150] trained on the COCO [147] detection dataset; symbol region embedding and attention (Eq. 5); adding symbol/object constraints (Eq. 6); and including additive knowledge (Eq. 7) using either KB objects or KB symbols.

Table 4: (Left) Ablation study on PSAs. All external knowledge components except attention improve over basic triplet embedding. (Right) Ablation on products. General-purpose recognition approaches, e.g. regions and attention, produce the main boost

Method	PSA				Product			
	Rank ↓	Rec@3 ↑	% improvement		Rank ↓	Rec@3 ↑	% improvement	
			Rank	Rec@3			Rank	Rec@3
base triplet	4.139	1.197			3.110	1.510		
generic region	3.444	1.375	17	15	2.650	1.670	15	11
symbol region	3.174	1.442	8	5	2.539	1.697	4	2
+attention	3.258	1.428	-3	-1	2.488	1.726	2	2
+symbol/object	3.149	1.466	3	3	2.469	1.727	1	<1
+KB objects	3.108	1.482	1	1	2.471	1.725	<1	<1
+KB symbols	3.013	1.509	4	3	2.469	1.725	<1	<1

The results are shown in Table 4 (left for PSAs, right for products). We also show percent improvement of each new component, computed with respect to the previous row, except for KB objects and KB symbols, whose improvement is computed with respect to the third-to-last row, i.e. the method on which both KB methods are based. The largest increase in performance comes from focusing on individual regions within the image. This makes sense because ads are carefully designed and multiple elements work together to convey the


WHEN THE WATER GOES, WE DRY.
OUROAZUL
New Ecofriendly Cutlery by GeoPack
GeoPack

VSE++: "I should try this makeup because its fun."

Hussain-ranking: "I should stop smoking because it destroys your looks."

ADVISE (ours): "I should be careful to how I treat Earth because when the water leaves we die."

VSE++: "I should wear Nivea because it leaves no traces."

Hussain-ranking: "I should be eating these because it has fresh ingredients."

ADVISE (ours): "I should buy GeoPack paper because their cutlery is eco-friendly."

Figure 9: Our ADVISE method compared to the two stronger baselines. On the left, VSE++ incorrectly guessed this is a makeup ad, likely because often faces appear in makeup ads. Hussain-Ranking correctly determined this is a PSA, but only our method was able to predict the topic, namely water/environment preservation. On the right, both Hussain-Ranking and our method recognized the concepts of freshness/naturalness, but our method picked a more specific statement.

message. We see that these regions must be learned as visual anchors to symbolic concepts (symbol region vs generic region) to further increase performance.

Beyond this, the story that the results tell differs between PSAs and products. Symbol/object constraints and additive branches are more helpful for the challenging, abstract PSAs that are the focus of our work. For PSAs, the additive inclusion of external information helps more when we directly predict the symbols (KB symbols), but also when we first extract objects and map these to symbols (KB objects). Note that KB symbols required 64,131 symbol labels. In contrast, KB objects relies on mappings between object and symbol words, which can be obtained more efficiently. While we obtain them as object-symbol similarities in our learned space, they could also be obtained from a purely textual, ad-specific resource. Thus, KB objects would generalize better to a new domain of ads (e.g. a different culture) where the data from [94] does not apply.

Qualitative Examples of the Symbolic Mappings. In Table 5, we show the object-symbol knowledge base that KB objects uses. We show "synonyms" across three vocabularies: the 53 symbol words from [94], the 27,999 words from the action/reason statements, and the 823 words from captions predicted for ads. We compute the nearest neighbors for each word in the learned space. This can be used as a "dictionary": If I see a given object, what should I predict the message of the ad is, or if I want to make a point, what objects should I use? In triplet ID 1, we see to allude to "comfort," one might use a soft sofa. From ID 2, if the statement contains "driving," perhaps this is a safe driving ad, where visuals allude to safety and injury, and contain cars and windshields. We observe the different role of "ketchup" (ID 3) vs "tomato" (ID 4): the former symbolizes flavor, and the latter health.

Table 5: Discovered synonyms between symbol, action/reason, and DenseCap words

ID	Symbol	Statement	DenseCap
1	*comfort*	couch, sofa, soft	pillow, bed, blanket
2	safety, danger, injury	*driving*	car, windshield, van
3	delicious, hot, food	*ketchup*	beer, pepper, sauce
4	food, healthy, hunger	salads, food, salad	*tomato*

In Fig. 10, we show the learned association between the individual words and symbolic regions. By learning from the ads image and statement pairs, our ADVISE model propagates words in the statement to the regions in the image thus associates each label-agnostic region proposal with semantically meaningful words (also the reason of using permutation invarient mean-pooling of texts). At training time, we have neither box-level nor word-level annotations.

Results on Additional Tasks. In Table 6, we demonstrate the versatility of our learned embedding, compared to the stronger two baselines from Table 3. None of the methods were retrained, i.e. we simply used the pre-trained embedding evaluated on statement ranking. First, we show a harder statement retrieval task: all statements that are to be ranked are from the same topic (e.g. all statements are about car safety or about beauty products). The second task uses creative captions that MTurk workers were asked to write for 2,000 ads in [94]. We rank these slogans, using an image as the query, and report the rank of the correct slogan. Finally, we check how well an embedding clusters ad images with respect to a ground-truth clustering defined by the topics of the ads.

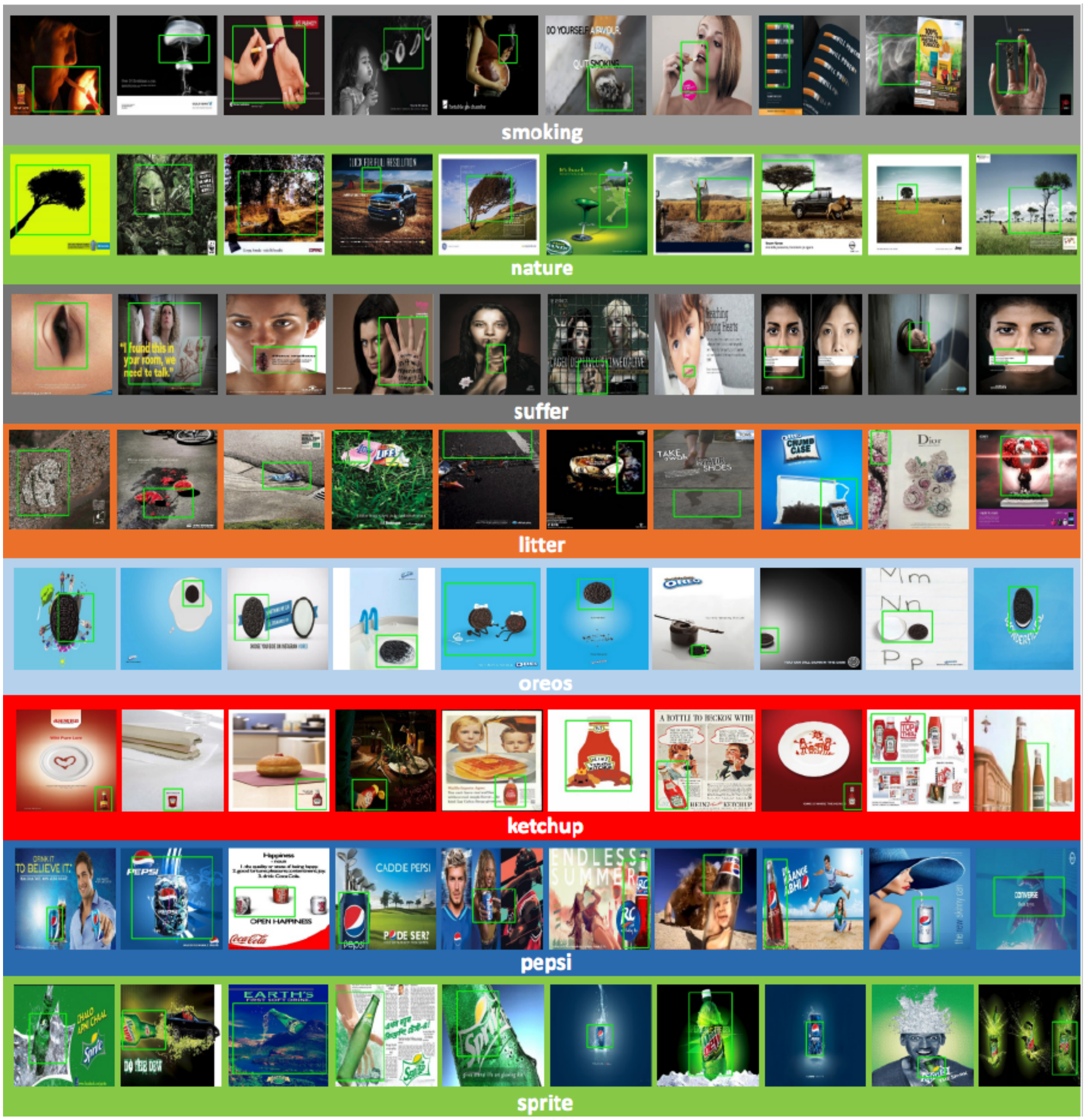


Figure 10: Learned association between the individual words and symbolic regions. We extract the CNN feature of each image region (Eq. 3), then use the word embeddings of "smoking", "nature", and so on to retrieve the most similar image regions (denoted using green boxes).

Table 6: Other tasks our learned image-text embedding helps with. We show rank for the first two (lower is better) and homogeneity [214] for the third (higher is better)

Method	Hard stmt. (↓ better)	Slogans (↓ better)	Clustering (↑ better)
Hussain-Ranking	5.595 (± 0.027)	4.082 (± 0.090)	0.291 (± 0.002)
VSE++	5.635 (± 0.027)	4.102 (± 0.091)	0.292 (± 0.002)
ADVISE (ours)	**4.827** (± 0.025)	**3.331** (± 0.077)	**0.355** (± 0.001)

3.3.2 Analysis of the Multimodal Features.

Baselines and Evaluation Metrics. To analyze the extensive multimodal features in Sec. 3.2.3, we use the splits defined in the ads challenge [123] (the challenge was organized after Sec. 3.2.2 was finalized). We use the 51,223 *trainval* images which are paired with 161,557 annotated statements for training; and evaluate on the 12,805 *test* images which are paired with 40,178 statements. We use TensorFlow [1] to build our model. We use a learning rate of 0.001, and the RMSProp optimizer with 0.95 decay and 1e-8 momentum. We use a batch size of 128, and all models are trained for roughly 60 epochs. To choose the best model, we use a held-out validation set with approximately 20% *trainval* data. For the similar action/reason ranking task on the *video data*, we split the 3,477 videos into trainval/test sets (80%/20%), resulting in 2,777 *trainval* and 700 *test* videos. We use a learning rate of 0.003 and roughly 170 epochs (3,000 steps). The remaining details are as for the image task.

We evaluate to what extent our proposed method (Sec. 3.2.3) is able to match an ad to its intended message; the message contains both the action and reason. We compare the following ablations of our method. All but the last one use an LSTM to encode the action-reason statements.

- Image Only uses region proposals trained from our symbolism data, without symbol labels: $L(\boldsymbol{z}, \boldsymbol{t}; \boldsymbol{\theta})$.
- Slogan Only is the method that uses OCR to extract the slogan embedded in the

image: $L(\boldsymbol{z}_{slg}, \boldsymbol{t}; \boldsymbol{\theta})$.

- Image+Slogan combines the image and slogan by optimizing $L(\boldsymbol{z} + \boldsymbol{z}_{slg}, \boldsymbol{t}; \boldsymbol{\theta})$.
- Image+Symbols uses additional knowledge from pretrained multi-label symbol classifier $\boldsymbol{u}_{symb}$, and optimizes $L(\boldsymbol{z} + \boldsymbol{z}_{symb}, \boldsymbol{t}; \boldsymbol{\theta})$.
- Full Method combines the region-based image representation, slogan, symbol and object: $L(\boldsymbol{z} + \boldsymbol{z}_{slg} + \boldsymbol{z}_{symb} + \boldsymbol{z}_{obj}, \boldsymbol{t}; \boldsymbol{\theta})$.
- Full Method (BOW) is the same as the previous method but uses bag of words representation, i.e. we average the individual word embeddings to get the full-text embedding (for statement, slogan, and object).

We evaluate the ablations in terms of two metrics: Accuracy which is the percentage of correct top-1 predictions; and Min Rank, which is the averaged ranking value of the best-ranked true matching statement (best possible rank is 1). We expect a good model to have high Accuracy and low Min Rank scores. We show results separately for product and public service announcement (PSA) ads, as in [294].

Results on the Image Ads. The results are shown in Tab. 7. The most important two modalities, as we expected, are the image (Image Only) and slogan (Slogan Only). Adding symbols to the image representation helps for products, but less than adding the slogan. Interestingly, the slogan extracted from the image seems to be more helpful than the image itself, likely because the slogan is more straight-forward and the image is designed to be attractive and creative, thus may be ambiguous in isolation. The fusion of both image and slogan modalities (Image+Slogan) outperforms both Image Only and Slogan Only. Image+Slogan outperforms Image Only by 34% on Product ads and 45% on PSAs, in terms of accuracy. This greater improvement on PSAs might be because these are less intuitive and more "clever" than product ads, thus hints from the slogan are more important. The inclusion of objects and symbols in our full method (Full Method) improves the accuracy of Image+Slogan on PSAs by 3%. Finally, aggregating all the text information using averaging (Full Method (BOW)) provides similar but slightly worse results compared to Full Method, i.e. accuracy reduced by 2% on product ads and PSAs.

We next break down the action-reason ranking task into two tasks, separately ranking the action and the reason. We do this in order to investigate which task is harder, and how much

Table 7: Ranking action-reason statements for image ads; high accuracy and low rank is desired. **Bold** is best, *Italics* is second- and third-best.

Method	Accuracy		Min Rank	
	Product	PSA	Product	PSA
Image Only	0.630	0.491	1.836	2.214
Slogan Only	0.791	0.677	1.599	1.788
Image+Slogan	**0.847**	*0.712*	*1.320*	*1.635*
Image+Symbols	0.640	0.489	1.764	2.241
Full Method	**0.847**	**0.733**	**1.282**	**1.554**
Full Method (BOW)	*0.827*	*0.718*	*1.318*	*1.588*

Table 8: Ranking action and reason statements separately, vs action-reason together. All methods shown use BOW. Numbers denote Min Rank (lower is better).

Method	Action-Reason	Action	Reason
Image Only	1.755	2.007	2.157
Slogan Only	1.532	1.758	1.845
Image+Slogan	**1.300**	**1.521**	**1.696**

observing the image or slogan contributes. The results are shown in Tab. 8. In general, the task of ranking the combined action-reason is the easiest one since it only requires the model to be confident about either the action or the reason. The additional image information (Image+Slogan vs Slogan Only) gives 18% reduction in rank while the extra slogan message (Image+Slogan vs Image Only) reduces rank by 35%. The action statement ranking is the second-easiest. Using the image gives 16% performance gain over slogan only. Predicting the reason statement is the most challenging, and offers the most limited room for improvement when using the image (9% over slogan only).

Results on the Video Ads. We show the performance on ranking statements for video ads. We use the same metrics and compare the following methods:

- Frame Only (BOF) is the model that only uses the video representation: $L(\boldsymbol{z}_{vdo}, \boldsymbol{t}; \boldsymbol{\theta})$.
- Speech Only (BOF) only uses the text information extracted by speech recognition: $L(\boldsymbol{z}_{spch}, \boldsymbol{t}; \boldsymbol{\theta})$.
- Frame+Speech (BOF) combines the bag-of-frames encoded video and speech by optimizing $L(\boldsymbol{z}_{vdo} + \boldsymbol{z}_{spch}, \boldsymbol{t}; \boldsymbol{\theta})$. This is our final model.
- Frame+Speech (LSTM) is similar but uses LSTM to encode all modalities (video, speech, and statement).

Table 9: Ranking action-reason statements for video ads. A good model has high accuracy and low rank. **Bold** is best, and *Italics* is second-best.

Method	Accuracy	Min Rank
Frame Only (BOF)	0.560	*2.401*
Speech Only (BOF)	0.507	2.987
Frame+Speech (BOF)	**0.639**	**2.053**
Frame+Speech (LSTM)	*0.561*	2.547

The results are shown in Tab. 9. Unlike the scenario in the image task, the directly detected spoken language (Speech Only (BOF)) is less useful than the pure visual cue (Frame Only (BOF)); the visual feature is 10% better in terms of accuracy. This implies that the ads designers did not put many unambiguous explanations in the conversation. However, we see that the conversation does help improve understanding, when used in combination with the frames: in terms of accuracy, Frame+Speech (BOF) is 14% better than Frame Only (BOF). Finally, we compare the BOF approach to a more complex model with more learnable parameters (Frame+Speech (LSTM)). We see that this latter models is 12% worse than the simpler BOF version. We surmise that the reason is the limited size of the video set. Note that when we use an LSTM to represent the visual information, results are similar to the BOF version; the drop comes from the LSTM representation of the *text* information.

3.4 Conclusion

We presented a method for matching image advertisements to statements which describe the idea of the ad. Our method uses external knowledge in the form of symbols and predicted objects in two ways, as constraints for a joint image-text embedding space, and as an additive component for the image representation.

For the thesis topic, we validated the hypotheses H1 and H3 (see Tab. 10). Thanks to the external knowledge and the attention building on region proposals (H1), our method outperforms existing image-text embedding techniques [53, 56] and a previous ad-understanding technique [94] by a large margin. The region embedding relying on visual symbolic anchors greatly improves upon traditional embeddings. For PSAs, regularizing with external info provides further benefit. Besides, the proposed constraints (supervisions) bridge the visual regions and the concepts in the knowledge bases. Thus, the learned model understands the symbolic meanings of the region proposals (H3).

Besides, we tested our image and video models combining multimodal channels on the tasks of predicting what the viewer should do and why. On the image ads dataset, we observed more than 30% improvements (32% for Product Ads and 50% for PSAs). Our method also achieved 14% improvements in video ads compared to the methods without using speech.

Table 10: Conclusion - validated hypotheses in this chapter.

	Multimodal features help to understand images/videos with implicit persuasive intent, such as visual advertisements.	Text features can be unreliable if not modeled appropriately.	Text supervision contains noise, but can be used to localize visual objects in space, if modeled properly.	Text supervision provides contexts regarding visual objects, they are reliable cues for disambiguating entities and relations.	Noisy audio narrations as a multimodal signal can be modeled to localize video actions in temporal domain.
This Chapter	✓		✓		

4.0 Story Understanding in Video Advertisements

4.1 Introduction

Different from the images, videos also involve temporal dynamics. To tell a full story using video media, not every second is presenting equally important information. Besides, video media are usually human-centric; thus, the multimodal features of human emotion, human motion, and so on, should be considered to understand the implicit intent.

In this chapter, we focus on videos and try to understand the dynamic structure of a video ad. As a powerful tool for affecting public opinion, video advertisements appeal to the viewers' emotions [304]. To achieve persuasive power, many ads explore creative narrative techniques. One classic technique is "Freytag's pyramid" where a story begins with exposition (setup), followed by rising action, then climax (action and sentiment peak), concluding with denouement or resolution (declining action) [60].

We model the dynamic structure of a video ad in this chapter. We track the pacing and intensity of the video, using both the visual and audio domains. We model how emotions change over the course of the ad. We also model correlations between specific settings (e.g., child's bedroom), objects (e.g., teddy bear) and sentiments (e.g., happy). We propose two methods to predict *climax*, "the highest dramatic tension or a major turning point in the action" [169], of a video. Then we use them along with rich context features to predict the *sentiment* that the video provokes in the viewer. Our framework is illustrated in Fig. 11. Our techniques are based on the following two hypotheses which we verify in our experiments.

First, we hypothesize that the climax of a video correlates with dramatic visual changes or intense content. Thus, we compute optical flow per frame and detect shot boundaries, then predict that climax occurs at those moments in the video where peaks in optical flow vectors or shot boundary changes occur. To measure dynamics in the audio domain, we extract the amplitude of the sound channel and predict climax when we encounter peaks in the amplitude. In addition to this unsupervised approach, we also show how to use the cues we develop as features, to predict climax in a supervised way. Both the unsupervised and supervised approaches greatly outperform the baseline tested.

Second, we hypothesize that video ads exploit associations that humans make, to create an emotional effect. We aim to predict the sentiment that an ad provokes in the viewer, and hypothesize that the setting and objects in the ad are greatly responsible for the sentiment evoked. We first extract predictions about the type of scene and type of objects in the ad, for each frame. We also hypothesize that the facial expressions of the subjects of the ad (i.e., the people in the ad) correlate with the sentiment provoked in the people watching it, so we also extract per-frame facial expression predictions. We treat sentiment prediction as a recurrent prediction task based on the scene, object, and emotion features, as well as features related to climax and standard ResNet [82] visual features.

To train our methods and test our hypotheses, we crowdsource climax annotations on 1,149 videos from the Video Ads Dataset of [94], and use the sentiment annotations provided.

To summarize, our contributions are as follows:

- We gather climax annotations for 1,149 video ads.
- We model the correlations between the climax and dramatic visual, audio changes.
- We show that video ads use human expressions, video scenes to create emotional effects.
- We show how to combine the visual and audio cues to predict the climax and sentiment.
- Our video ads sentiment prediction model outperforms the only prior work [94] by a large margin.

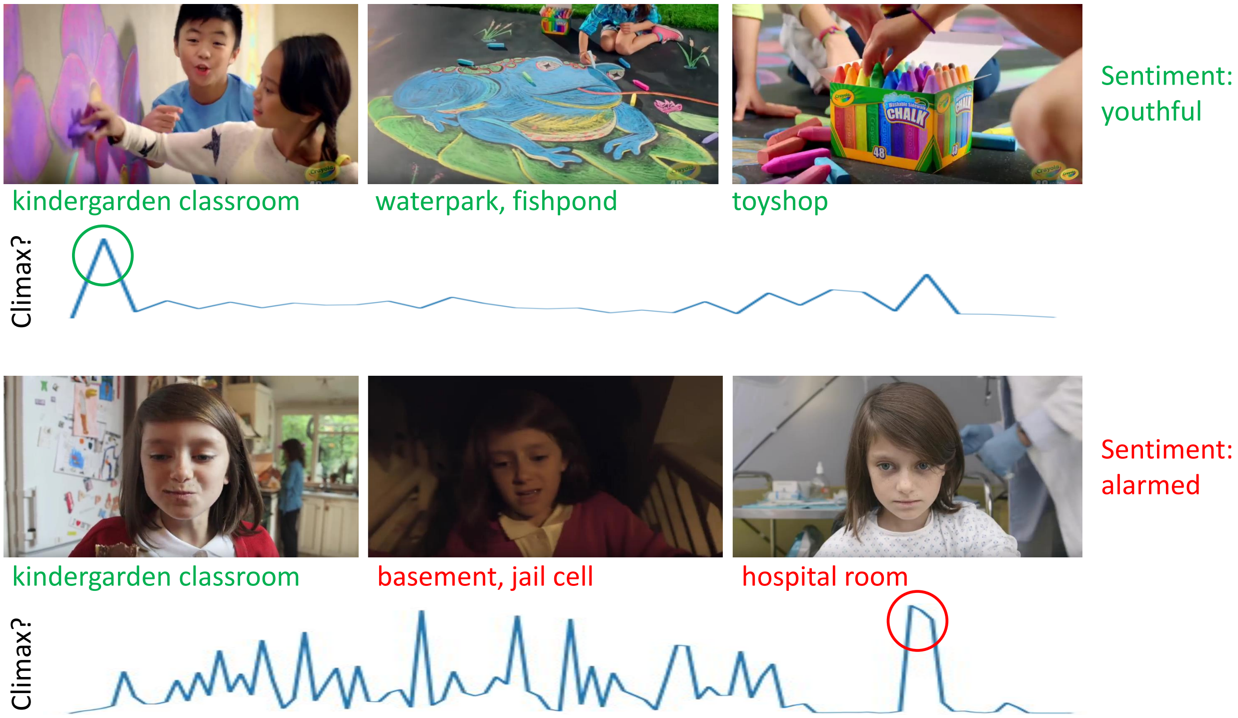


Figure 11: The key idea behind our approach. We want to understand the story being told in the ad video, and the sentiment it provokes. We hypothesize that the semantic content of each frame is quite informative and that we need to model the rising action to understand which temporal parts most contribute to the sentiment. We show the places recognized in the frames of two videos, as well as soft predictions about whether a certain frame corresponds to the climax of the video or not. While both videos start with images of children, which might indicate positive sentiment denoted in green (e.g. "youthful"), this positive trend only remains in the first video (indicated by places correlated with youthfulness, such as "toy shop"). In contrast, the second video changes course and shows unpleasant places (denoted in red) e.g. "basement" and "hospital room". Because the climax in the second video occurs near the end, our method understands that it is these later frames that determine the sentiment ("alarmed").

4.2 Approach

In The Advertising Research Handbook [304], dramatic structure has four prototypical forms, shown in Fig. 12 (based on [304] p.212). These structures depend on how positive and negative sentiment rises or declines. [304] examines product ads, and the changes in positive/negative sentiment are correlated with appearances of the brand. In public service announcements (PSAs), the role of positive/negative might be reversed, as PSAs often aim to create negative sentiment in order to change a viewer's behavior. However, understanding the story of PSAs still depends on understanding the climax of (negative) sentiment. Thus, we first collect data (Sec. 4.2.1) and develop features (Sec. 4.2.2) that help us predict when climax occurs. We then develop features informative for sentiment (Sec. 4.2.3). We finally describe how we use these features to predict the type of sentiment and occurrences of climax (Secs. 4.2.4 and 4.2.5).

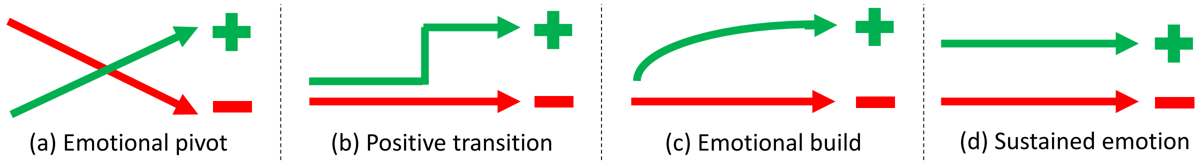


Figure 12: The "four archetypes of dramatic structure" in product ads [304] which motivate our approach. For PSAs, the roles of positive and negative sentiments might be reversed.

4.2.1 Climax and Sentiment Data

We use the Video Ads Dataset of [94]. It contains 3,477 video advertisements with a variety of annotations, including the sentiment that the ad aims to provoke in the viewer. We collected climax annotations on a randomly chosen subset of 1,595 videos from this dataset, using the Amazon Mechanical Turk platform. We restricted participation on our tasks to annotators with at least 98% approval rate who submitted at least 1000 approved tasks in the past. We submitted each video for annotation to four workers. Each was asked to watch the video and could choose between two options, "the video has no climax" or "the video has climax." If the latter, the worker was asked to provide the minute and second at which climax occurs (most videos are less than 1 min long). To ensure quality, annotators were also asked to describe what happens at the end of the video. Some of the videos in [94]'s dataset were not available, so the annotators could also mark this option. We ended up with 1,149 videos that contain climax annotations. We manually inspected a subset of them and found the timestamps were quite reasonable. The descriptions of what happened at the end were often quite detailed. We will make this data publicly available upon publication.

4.2.2 Climax Indicators

We first analyze the dynamics of the video, using both visual and audio channels. We plot time on the x-axis, and measurement of dynamics/activity on the y-axis (Fig. 13). We consider three indicators of rapid activity: the amplitude of audio signals, the occurrence of shot boundaries, and the magnitude of optical flow vectors between frames.

In particular, we extract these features and portray them as follows:

- **The audio amplitude $\boldsymbol{a}^k$**, which is the max amplitude of audio for the k-th frame. We first extract the sound channel from the video, take a fixed number of samples from the sound wave per second, then compute the max across the samples for that frame.
- **The shot boundary indicator**, which is equal to 0 or 1 depending on whether a shot boundary occurs in the k-th frame. We use [29] for shot boundary extraction. In order to obtain more informative cues, we vary the parameters of [29] to get five 0/1 predictions per frame and use this 5D prediction $\boldsymbol{b}^k$ as the representation for the k-th frame. To generate the plot in Fig. 13, we aggregate information over all frames in a given second.
- **The optical flow magnitude $\boldsymbol{o}^k$**, which is computed as $\frac{1}{W*H}\sum_{i=1}^{W}\sum_{j=1}^{H}\sqrt{(u_{i,j}^k)^2+(v_{i,j}^k)^2}$ where $u_{i,j}^k$ and $v_{i,j}^k$ are the horizontal and vertical optical flow components for each pixel (i, j) in the k-th frame. We use [207] to extract optical flow vectors.

4.2.3 Sentiment Indicators

The Advertising Research Handbook [304] describes the dramatic structure of ads as closely depending on the emotion of the video. One type of structure (Fig. 12) is the "emotional pivot" where an ad starts with negative sentiment, which declines over time, to make

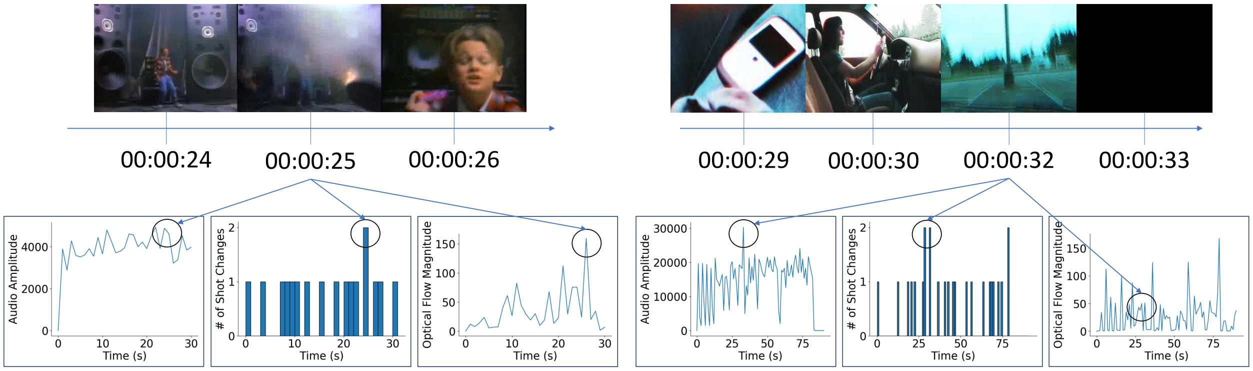


Figure 13: The audio, shot boundary frequency, and optical flow plots for two videos, along with frames from the videos corresponding to climactic points. The first video shows an "explosion" around the 25th second, and the second shows a car crash around the 32nd second. The circles correspond to the timestamp of the frames shown. In the first video, climax is detected well in each of the three plots. In the second, shot boundaries and audio are informative, but optical flow is not.

room for increasing positive sentiment. The "emotional build" involves a gradual increase and climax in positive sentiment. Thus, the sentiment is equally crucial to understanding the story of the ad video as the climax. Since an ad targets an audience and wants to convince the audience to do something, it is the viewer's sentiment that matters the most.

[94] contains annotations about what sentiment each ad video provokes in the viewer, collected from five annotators. These annotations involve 30 sentiments, both positive (e.g., cheerful, inspired, educated), negative (e.g., alarmed, angry) and neutral (e.g., empathetic). [94] also includes a baseline for predicting sentiment, using a multi-class SVM and C3D features [257]. The authors extract features from 16-frame video clips, then average the features. Thus, their model does not capture the dynamics and sequential nature of the video. We hypothesize that if we model how the content of the video changes *over time*, and consider the *context* in which the sentiment in the video is conveyed, we would be able to model sentiment more accurately. We model sentiment with the following intuitive context features:

- **The setting in each frame of the video**, i.e. the type of place/scene. Let $\boldsymbol{vp} = \{p_1, \ldots, p_{365}\}$ be the vocabulary of places in the Places365 dataset [325]. We use a pre-

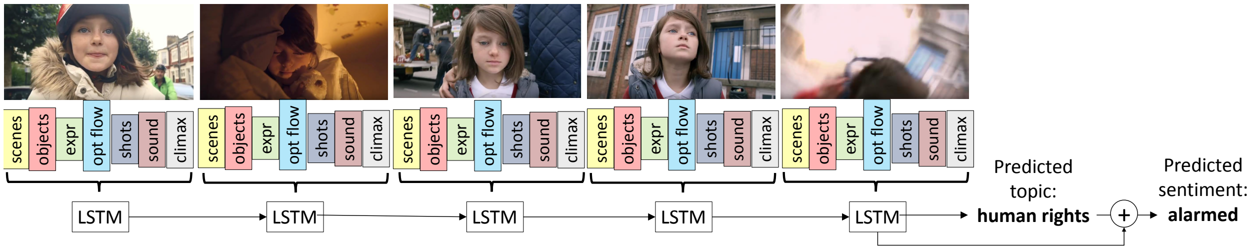


Figure 14: Our dynamic context-based approach. The last frame shows an explosion.

trained prediction model from [325] to obtain a 365D vector $\boldsymbol{pl}^k = [l_1^k, \ldots, l_{365}^k]$, where l_i^k is the probability that the k-th frame exemplifies the i-th place.

- **The objects found in the video**. Let $\boldsymbol{vo} = \{c_1, c_2, \ldots, c_{80}\}$ be the vocabulary of the COCO object detection dataset [147]. We use the model of [89] trained on COCO to get the objects in a frame. We then use max-pooling to turn the detection results into an 80D fixed-length feature vector $\boldsymbol{ob}^k = [s_1^k, \ldots, s_{80}^k]$, where s_i^k is the maximum confidence score among multiple instances of the same object class c_i, in frame k.
- **The facial expressions in the video**. We observed that the overall sentiment that the video provokes *in the viewer* often depends on the emotions that the *subjects* of the video go through. For example, if a child in an ad video is initially "happy" but later becomes "sad," the sentiment provoked in the adult viewer might be "alarmed" because something disturbing must have happened. Thus, we also model emotions predicted on faces extracted per frame. We first detect the faces using OpenFace [6]. We then extract the expression of each face using an Inception model [244] trained on the AffectNet dataset [178]. Two types of results are predicted: (1) the probability distribution among the eight expressions defined in AffectNet, and (2) the valence-arousal values for the face, saying how pleased and how active the person is (in range -1 to +1). We average the face expressions (10 values) for all faces detected in the k-th frame, to get the 10D final representation $\boldsymbol{fa}^k$.
- **The topic of the ads**. [94] defines a vocabulary of 38 topics in the ads domain and also provides annotations for these topics. We hypothesize the overall sentiment that the video provokes is related to the topic the ad belongs to. For example, "sports" ads usually convey "active" and "manly" sentiments, while "domestic violence" ads often

make people feel “sad”. Thus, we designed a multi-task learning framework with two objectives: one for the topic and the other for the sentiment prediction, hoping the topic prediction can help the prediction of sentiments. We first use the video-level feature (the last hidden state of the LSTM) to predict the 38D topic distribution, then concatenate this 38D vector with the video-level feature to predict the sentiment. The idea is described in Fig. 14.

- **The video frame-level CNN features**. We also use features from the last layer of a ResNet trained on ImageNet [82, 215].

4.2.4 Unsupervised Climax Prediction

We can directly predict that climax occurs at times which are peaks in terms of shot boundary frequency, optical flow magnitude, or audio amplitude. Since the shot boundary frequency can be the same for many timeslots, we look for the longest sequence of timeslots which contain at least one shot boundary and predict the center of this “run” as a peak. Optical flow magnitudes and audio amplitudes are compared on a second-by-second basis. We extract the top-k maximal responses from each plot, predict these as climax, and evaluate the performance in Sec. 4.3.3.

4.2.5 Supervised Climax/Sentiment Prediction

We predict climax using an LSTM (with 64 hidden units) that outputs 0/1 for each frame, where 1 denotes that the frame is predicted to contain climax. The frame-level features used are ResNet features (2048D), optical flow magnitude $\boldsymbol{o}^k$ (1D), the shot boundary indicator $\boldsymbol{b}^k$ (5D), the sound amplitude $\boldsymbol{a}^k$ (1D), the place representation $\boldsymbol{pl}^k$ (365D), the object representation $\boldsymbol{ob}^k$ (80D), and the facial expression feature $\boldsymbol{fa}^k$ (10D).

For the sentiment prediction task, we also use an LSTM with 64 hidden units. We use the same frame-level features as the climax prediction. Moreover, we also add the predicted climax (1D) as extra information. Ads topics are used as both an additional loss/constraint and an extra feature for the sentiment prediction (see Fig. 14).

Discussion. The advantages of our approach are as follows. First, the distribution of object, place, and facial expression probability vectors is much lower-dimensional than ResNet features, so given the limited size of the Video Ads Dataset (3,477), formulating the problem as learning a mapping from objects/scenes/facial expressions to sentiments/climax is much more feasible. The optical flow, shot boundary, and sound features are also very low-dimensional, and have clear correlation with the presence of climax. Further, understanding the sentiment of a video and its climax are related tasks. Thus, it is intuitive that climax predictions should be allowed to affect sentiment prediction; this is the idea shown in Fig. 11 where we use climax to select the part of the video which affects the elicited sentiment the most. We show in Sec. 4.3.4 (Table 14) that our semantic/climax features outperform the ResNet features, and the combination of the two achieves the strongest performance.

4.3 Experiments

We first describe our experimental setup and training procedure, then present quantitative and qualitative results on the climax and sentiment prediction tasks.

4.3.1 Evaluation Metrics

For the climax prediction task, we use the recall of the top-k prediction ($k = 1, 3$) to measure performance. Since exactly matching the ground-truth climax timestamp is challenging, we apply an error window saying that the prediction is treated as correct if the ground-truth climax is close (within $0, 1, 2$ sec). We treat the prediction as correct if it recalls any of the ground-truth annotations for that video, except rejected work. Table 11 shows the results.

To measure how well the model's prediction agrees with the sentiment annotations, we compute mean average precision (mAP) and top-1 accuracy (acc@1) based on three forms of agreement (agree with k, where $k = 1, 2, 3$). "Agree with k" means that we assign a ground-truth label to a video only if at least k annotators agree on the existence of the sentiment.

The acc@1 is the fraction of correct top-1 predictions across all videos, and the mAP is the mean of the average precision over evenly spaced recall levels. Tables 12, 13 and 14 show the results.

4.3.2 Training and Implementation Details

For training both the climax and sentiment prediction models, we use the TensorFlow [1] deep learning framework. We split the Video Ads Dataset [94] (3,477 videos) into train/val/test (60%/20%/20%), resulting in around 2,000 training examples for the sentiment prediction task and about 700 training examples for the climax prediction task (since only 1,149 of the 3,477 videos have climax annotation). We report our results using five-fold cross-validation.

For the climax prediction task, we use a one-layer LSTM model with 64 hidden units. At each timestamp, the model predicts a real value ranging from $[0, 1]$ (output of the sigmoid function) denoting whether the corresponding frame contains a climax. We then use the sigmoid cross entropy loss to constrain the model to mimic the human annotations. Considering the size of the dataset, we set both the input and output dropout keep probability of the LSTM cell to 0.5 to avoid over-fitting. We use the RMSprop optimizer with a decay factor of 0.95, momentum of 1e-8, and learning rate of 0.0002. We train for 20,000 steps using a batch size of 32, and we use the recall of the top-1 prediction (the error window is set to "within 2 seconds") to pick the best model on the validation set.

For the sentiment prediction task, we use the same procedure, but we pick the best model using mAP using "agreement with 2". We use the last hidden state of the LSTM to represent the video feature and add a fully connected layer upon it to get the 38D topic representation. We then concatenate the 38D topic representation with the last hidden state of the LSTM and infer a 30D sentiment logits vector from the concatenated feature. The sigmoid cross entropy loss is also used here. Similar to [251], we found that using soft scores as ground-truth targets improves the performance and makes the training more stable. To deal with data imbalance for the rare classes, we sampled at most $5n$ negative samples if there were n positives.

4.3.3 Results on Climax Prediction

We show the results of unsupervised and supervised climax prediction in Table 11. We measure whether the predicted climax is within 0, 1, or 2 seconds of the ground-truth climax. We first show a heuristic-guess baseline which always predicts that climax occurs at 5 seconds for the top-1 prediction and at 5, 15 and 25 seconds for top-3. We then show the performance of the three unsupervised climax prediction methods described in Sec. 4.2.4. Next, we show the performance of 0/1 climax prediction (Sec. 4.2.5) using an LSTM with ResNet features only, and finally our method using the features we proposed in both Sec. 4.2.2 and Sec. 4.2.3 (excluding the video-level topic feature).

Table 11: Climax prediction with best performer per setting in **bold** and second performer in *italics*. Unsupervised prediction performs quite well. Our supervised method achieves the best or second-best performance for all settings. For the "LSTM, ResNet only" approach, we guess the reason that it is competitive is that LSTM has the ability to capture the temporal dynamics to a certain degree

Method	top-1 prediction			top-3 prediction		
	w/in 0 s	w/in 1 s	w/in 2 s	w/in 0 s	w/in 1 s	w/in 2 s
baseline	0.031	0.083	0.121	0.122	0.299	0.430
shot boundary (unsup)	0.068	0.179	0.265	*0.221*	**0.457**	**0.588**
optical flow (unsup)	0.064	0.152	0.220	0.163	0.380	0.513
audio (unsup)	**0.077**	0.171	0.255	0.178	0.403	0.534
LSTM, ResNet only	0.071	*0.206*	**0.290**	0.190	0.400	0.523
LSTM, all feats (Ours)	**0.077**	**0.209**	*0.287*	**0.226**	*0.439*	*0.546*

We see that the unsupervised methods, and especially shot boundary and audio, greatly outperform the baseline. Interestingly, audio performs quite well in the hardest setting, only one shot at prediction and exact alignment between predicted and ground-truth climax. Shot boundary achieves the best performance in the two weakest settings (top-3 predictions, agreement within 1-2 seconds). In all settings, our method achieves the best or second-best performance.

4.3.4 Results on Sentiment Prediction

Table 12 shows our main result for sentiment prediction. We compare to Hussain et al. [94]'s method which is a multi-class SVM model using the C3D features [257]. This is the only prior method that attempts to predict sentiment on the Video Ads Dataset. We observe that our method improves upon [94]'s performance for most metrics. The improvement is more significant for mAP, which is more reliable because of the imbalance of the dataset. We improve the mAP compared to prior art by up to 25% in terms of agreement with 3 annotators. For reference, human annotators' agreement with 1 (at least one other annotator) is 0.723.

Table 12: Our method outperforms prior art for sentiment prediction

Method	Agree with 1		Agree with 2		Agree with 3	
	mAP	acc@1	mAP	acc@1	mAP	acc@1
Hussain et al. [94]	0.283	0.664	0.135	0.435	0.075	**0.243**
Our model	**0.313**	**0.712**	**0.160**	**0.449**	**0.094**	0.241

Table 13 examines the contribution of the features described in Sec. 4.2.2 and Sec. 4.2.3, and the use of an LSTM to model dynamics of the video. We compare against an LSTM that uses only ResNet features. We also compare to a bag-of-frames (BOF) method that rules out the effects of dynamics. It computes the final video-level representation by simply applying mean pooling among the frame-level features. We observe that our method (using the proposed features and LSTM) always outperforms the other methods in terms of mAP scores. Our method achieves significant improvement over the second-best method (10% for mAP and agreement with 2, and 21% for mAP and agreement with 3). In terms of accuracy, all methods perform similarly, and the best model (BOF, all features) also uses our proposed features.

Table 14 verifies the benefit of each of our features. We show the LSTM-ResNet-only

Table 13: In-depth evaluation of the components of our method for sentiment prediction

Method	Agree with 1		Agree with 2		Agree with 3	
	mAP	acc@1	mAP	acc@1	mAP	acc@1
BOF, ResNet only	0.295	0.708	0.141	0.449	0.076	0.242
LSTM, ResNet only	0.302	0.716	0.145	0.451	0.074	0.242
BOF, all features (incl. ours)	0.302	**0.719**	0.146	**0.462**	0.078	**0.248**
LSTM, all features (Our model)	**0.313**	0.712	**0.160**	0.449	**0.094**	0.241

baseline from Table 13, then eight methods which add one of our features at a time, on top of this baseline. Next, we show an LSTM method which uses our features without the base ResNet feature, and finally, our full method. We use mAP for agreement with 3 in the table. We show the average result across all sentiment classes, then results for four individual ad sentiments. In bold are all methods which improve upon the ResNet baseline. We see that all of our features (the average column) contribute to the performance of our full method. Using all features except ResNet is stronger than using ResNet features alone. We note models based on individual features still show benefits on specific sentiment classes, and we believe the reason is that our fusion method is too simple to aggregate all the information.

We observe some intuitive results for the four chosen individual sentiments. We ranked sentiments by frequency in the dataset and picked the 6th, 7th, 9th and 13th most frequent. For “educated,” the places feature is most beneficial, which makes sense because “education” might occur in particular environments, e.g., classroom. As shown in our example ad in Fig. 14, the setting (e.g., places) and dramatic content changes (measured by optical flow and shot boundaries) are quite telling of the “alarmed” sentiment. Most features help greatly for the “fashionable” sentiment. For “angry”, audio is very helpful (43% improvement over ResNet), which makes sense since loud speaking might trigger or correlate with anger.

We show qualitative examples in Fig. 15. Our model using the extensive features correctly predict “amazed” and “fashionable” while the baseline method does not. Our method relies

Table 14: Ablation study evaluating the benefit of each feature for sentiment prediction, using agreement with 3 mAP. In bold are all methods that outperform the baseline

	average	educated	alarmed	fashionable	angry
ResNet only (baseline)	0.074	0.036	0.117	0.047	0.007
objects	**0.082**	0.032	**0.140**	**0.080**	0.004
places	**0.082**	**0.074**	**0.132**	**0.160**	0.005
facial expressions	**0.077**	**0.044**	**0.143**	**0.084**	0.003
topic	**0.086**	0.032	**0.143**	**0.136**	**0.009**
optical flow	**0.082**	**0.045**	**0.150**	**0.133**	0.005
shot boundaries	**0.080**	**0.037**	**0.151**	**0.110**	0.003
audio	**0.077**	**0.040**	0.113	**0.116**	**0.010**
climax	**0.079**	0.025	**0.119**	**0.082**	**0.011**
all features except ResNet	**0.080**	**0.038**	0.104	0.036	0.007
all features (Our model)	**0.094**	0.026	0.099	**0.202**	0.005

on recognized places (e.g. laboratory, beauty salon), objects, facial expressions, and climax dynamics.

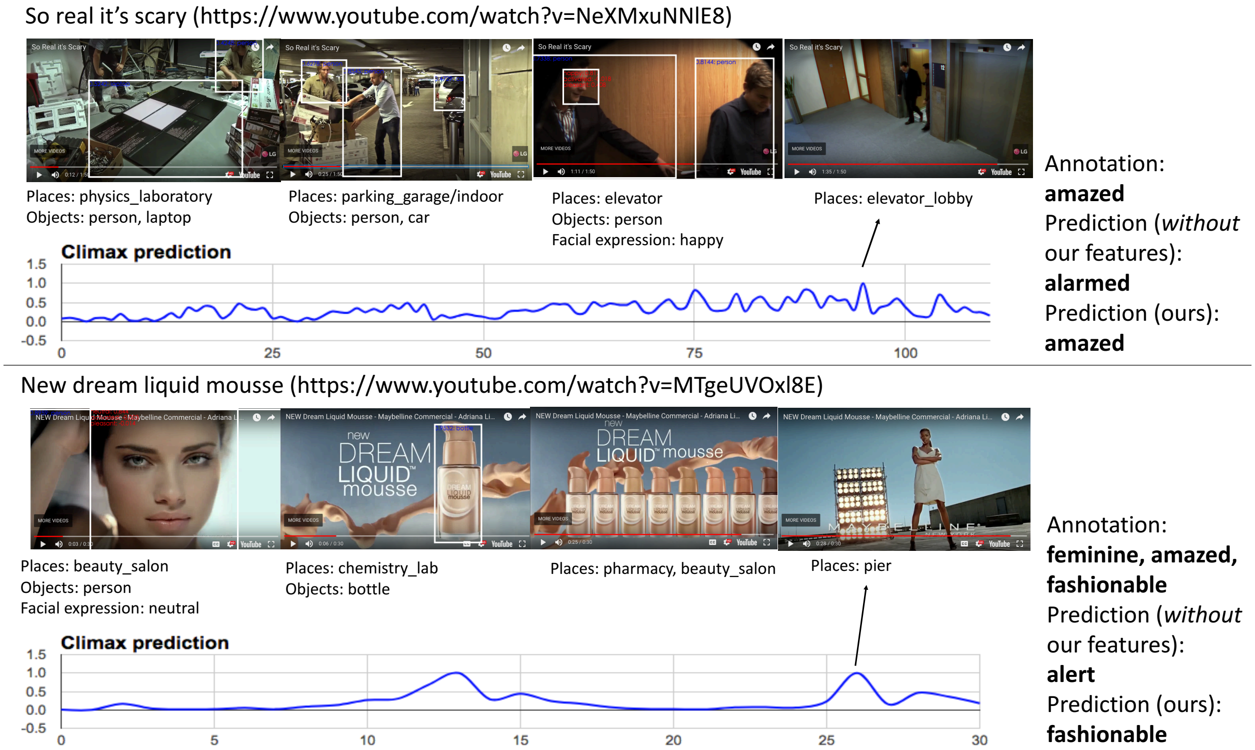


Figure 15: Qualitative results from our model. More examples could be found at `http://people.cs.pitt.edu/~yekeren/ads_climax/demo/`

4.4 Conclusion

We made encouraging progress in understanding the dynamic structure of a video ad. We hypothesized that climax correlates with dramatic visual and audio changes. We crowdsourced climax annotations on 1,149 videos from the Video Ads Dataset of [94] and used both unsupervised and supervised methods to predict the climax.

We proved the thesis hypotheses H1 and H5 (see Tab. 15). We show that the multimodal cues have strong correlations to the climax (H5); hence, we can use them to localize climax in an unsupervised manner. By combining visual and audio cues with semantically meaningful context features (H1), our sequential model (LSTM) outperforms the only prior work [94] by a large margin, on the sentiment prediction task. To better understand the relations between the semantic visual cues and the sentiment each ad video provokes, we performed detailed ablations and found all the features we proposed help to understand the evoked sentiment.

From the next chapter, we study the way to efficiently utilize the multimodal features.

Table 15: Conclusion - validated hypotheses in this chapter.

	Multimodal features help to understand images/videos with implicit persuasive intent, such as visual advertisements.	Text features can be unreliable if not modeled appropriately.	Text supervision contains noise, but can be used to localize visual objects in space, if modeled properly.	Text supervision provides contexts regarding visual objects, they are reliable cues for disambiguating entities and relations.	Noisy audio narrations as a multimodal signal can be modeled to localize video actions in temporal domain.
This Chapter	✓				✓

5.0 Breaking Shortcuts by Masking for Robust Visual Reasonin

5.1 Introduction

Chapter 3, 4 discussed the use of multimodal features and external knowledge in ads understanding, a particular visual reasoning task. From this chapter, we start to extend our discussion beyond ads. We pay more attention to what visual reasoning is and how to use the multi-channel inputs for it efficiently. Although this chapter still uses the ads data, the proposed method of using bottom-up graph and stochastic masking is a generalizable machine reasoning process. Hence it can be adapted to additional data rather than ads. In the next chapter, we use a different Visual Commonsense Reasoning (VCR) dataset [315].

Visual reasoning is an important family of problems that are of increasing interest. Example problems in this space include visual question answering [12, 79, 91, 233] and visual commonsense reasoning [314] (see Fig. 16). The general approach is to learn a joint embedding space for images, questions and answers, then learn to generate or retrieve a correct answer, by minimizing a loss computed using supervised training data. The name "reasoning" bears a flavor of classic AI and structured logic-inspired inference steps; one might argue that a human accumulates knowledge as they mature, and they store this knowledge in a metaphorical "knowledge base", then retrieve information from it as needed. Though in many domains state of the art performance is achieved by end-to-end-trained transformer models [37, 152, 245], some approaches to VQA/VCR do rely on symbolic reasoning [9, 106, 261, 271], and they just like humans.

In this chapter, we design a bottom-up graph model (see Fig. 17) to mimic the human symbolic reasoning process and discuss a more general problem: how to use multimodal features and external knowledge efficiently. We test two solutions. On the one hand, we consider the weighting of the multiple input channels. Since some channels contain informative features and the others contain noises, we use weighting to adjust the reliance on them. On the other hand, not all co-occurrences of the feature-answer pairs uncover the natural rules (e.g., holding up an umbrella does NOT necessarily cause the rain). Thus, we propose

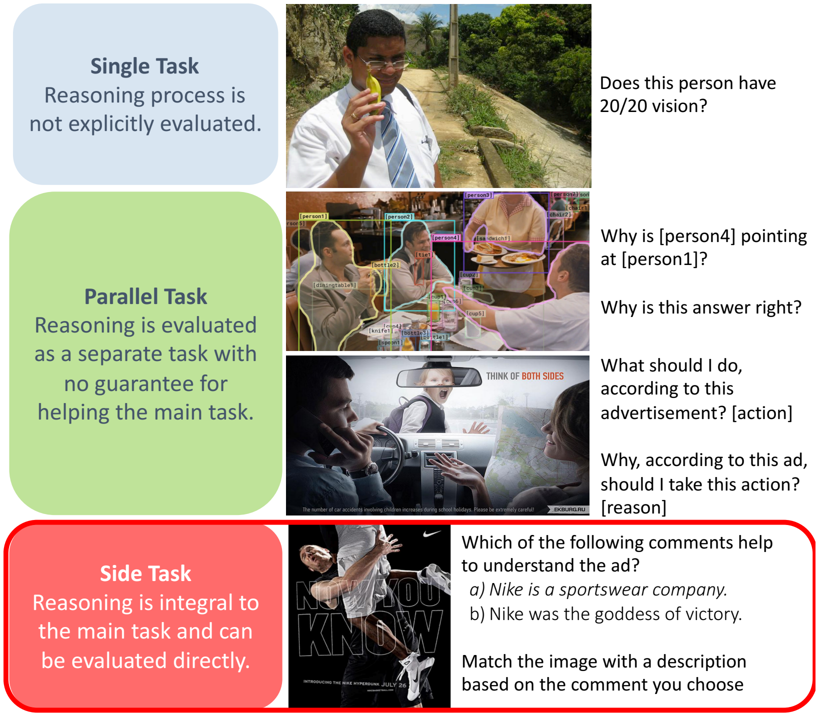

Single Task
Reasoning process is not explicitly evaluated.
Does this person have 20/20 vision?
Parallel Task
Reasoning is evaluated as a separate task with no guarantee for helping the main task.
Why is [person4] pointing at [person1]?
Why is this answer right?
What should I do, according to this advertisement? [action]
Why, according to this ad, should I take this action? [reason]
THINK OF BOTH SIDES
Side Task
Reasoning is integral to the main task and can be evaluated directly.
Which of the following comments help to understand the ad?
a) Nike is a sportswear company.
b) Nike was the goddess of victory.
Match the image with a description based on the comment you choose
NOW YOU KNOW

Figure 16: Visual understanding tasks. Previous definitions either oversimplified reasoning (as answering, see top) or treat reasoning as a standalone task parallel to answering (middle). In contrast, we propose a new evaluation side task (bottom) that checks the decisions made by our main model, i.e. which knowledge pieces it selected to complete the answering task.

a stochastic masking technique to break the unreliable co-occurrences occasionally.

Our key novelty is the idea that by randomly masking parts of the training data we can force the method to focus on more generalizable patterns. Specifically, we mask parts of text found in the image, of knowledge pieces, and of the answer options, with some probability. This technique prevents models from learning simple string-match or object-match between input and output.

To evaluate our model, we use a visual reasoning task for advertisement understanding. Given a visual ad, the method needs to retrieve the correct "action-reason" statement describing this image. Action-reason statements capture the *action* that the ad implies the viewer should take and *reasons* it provides for taking the suggested action (e.g. "I should buy these shoes because they will make me athletic", for the example in Fig. 17). Note that the word “reason” in this context is akin to “rationale”. In contrast, by “reason*ing*” we mean the ability to use the right evidence to select an action-reason statement.

We show that masking allows our model to improve the standard metrics used for evaluating advertisement understanding. However, we find these metrics are not sufficient to evaluate a model’s ability to reason using suitable evidence. Thus, we propose a new side task, used only for evaluation (not training). In this task, we verify whether the external knowledge that our method chose to use, is actually supporting the reasoning. To facilitate the evaluation, we collect human annotations on a small test set. We show that our method more than *doubles* the accuracy of the knowledge selection mechanism.

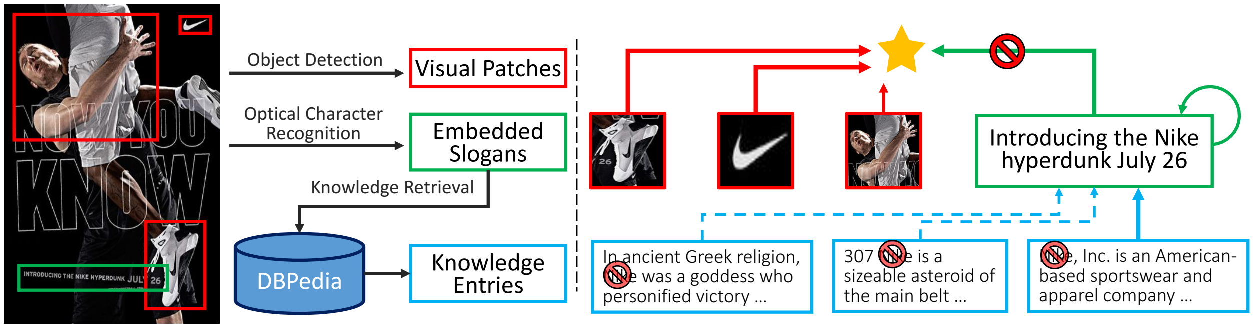

Object Detection
Visual Patches
Optical Character Recognition
Embedded Slogans
Knowledge Retrieval
DBPedia
Knowledge Entries
Introducing the Nike hyperdunk July 26
In ancient Greek religion, Nike was a goddess who personified victory ...
307 Nike is a sizeable asteroid of the main belt ...
Nike, Inc. is an American-based sportswear and apparel company ...

Figure 17: Overview of the proposed model. Given a single image ad, we first expand the representation using object detection and OCR, and also retrieve relevant knowledge based on slogan snippets (left). We build a graph-based model to infer the overall message using all available information (right). For more effective training, we mask query keywords and randomly drop certain knowledge pieces (shown with red crossed-out circles). More details are in Sec. 5.2.

To summarize, our contributions are as follows:

- a bottom-up graph model utilizing external knowledge,
- a method for more generalizble reasoning, by using masking of retrieved knowledge and image evidence, to prevent the model from learning shortcuts,
- a new side task (with annotations) to evaluate reasoning ability, and
- state of the art results on visual reasoning about advertisements [94, 294, 297].

5.2 Approach

We focus on one specific reasoning task, namely advertisement understanding. We incorporate image regions, text in the image, and external DBpedia knowledge [134], in a graph model. Because we retrieve knowledge from an open, general, real-world knowledge base, retrieved *irrelevant* pieces of knowledge dominate in count. We thus allow our model to select which pieces of knowledge and information to leverage, using learnable scalar edge weights.

One interesting but easy to neglect problem is that when the answer options can easily be matched to the image evidence, additional information (external knowledge) may not be necessary and hence may not help performance on the main task. We show a small example in VCR [314], where the subject repetition seemed to be the trick to answer the question without knowing the visual cues:

How is Jackie feeling? Avery is very excited.

How is Jackie feeling? Jackie is focused and active.

There is a similar problem in ads understanding. For example, given a Nike ad with an embedded slogan containing the word "Nike", the model must retrieve external knowledge to infer the particular properties that this ad demonstrates, so it can select the correct action-reason statement. However, the model can also find a shortcut and *not* perform reasoning, by merely looking for potential choices containing the brand name. Another example is the famous PepsiCo celebrity branding, where a naive model can simply *remember* popular celebrities and directly match them to "Pepsi" rather than understanding their shared characteristics (e.g. athleticism), thus it may generalize poorly if a new spokesperson is introduced in the ads. We refer to this observation as **shortcut learning**. Specifically, when there exists a superficial image-answer match on the surface, the model tends to lazily seek it and avoid squeezing more useful information out of the retrieved knowledge, even when abundant resources are available.

Below, we first describe the advertisement understanding task (Sec. 5.2.1). We then introduce our overall framework and how we train (Sec. 5.2.2). We describe our image representation (Sec. 5.2.3-5.2.4) and knowledge selection mechanisms (Sec. 5.2.5). Finally, we describe our strategy for breaking shortcuts and forcing the model to "study harder" and learn more generalizable patterns (Sec. 5.2.6).

5.2.1 Task: Advertisement Understanding

We focus on the advertisement understanding task [94] because it considers an interesting and practical scenario. First, ads exploit symbols that refer to content outside the image; thus, retrieving external knowledge is required. Second, unlike [225, 314], neither external knowledge nor reasoning rationales are available in clean form. Third, multiple modalities (image and slogan text) must be considered.

For each image, [94] provide three statements in which each is an action-reason pair (e.g., "I should buy Nike because it protects my feet."). There may be multiple plausible reasons per action, e.g. to buy "sportswear", the image may argue "it protects", "is cheap", or "celebrity wears it". Models are required to match an advertisement with the correct *action-reason* descriptive statement.

Given an ad image A, we assume it is composed of two parallel entity sets $A = \{V, T\}$, where V stands for visual signals and T represents the embedded slogans (i.e. textual signals). For each image, we apply off-the-shelf object detectors to generate a group of object proposals as the salient visual signals from the ad, noted as $V = \{v_1, v_2, \ldots, v_{|V|}\}$. We also use existing optical character recognition (OCR) engines to extract embedded text slogans as $T = \{t_1, t_2, \ldots, t_{|T|}\}$.

5.2.2 Training: Matching to the Statements

We follow the approach in [294] and use triplet loss (Eq. 8) to optimize the cosine similarity $\text{cosine}(\mathbf{h}, \mathbf{s}) = \dfrac{\mathbf{h} \cdot \mathbf{s}}{\|\mathbf{h}\| \cdot \|\mathbf{s}\|}$ between advertisement representation $\mathbf{h}$ and answer choice statement embedding $\mathbf{s}$. Eq. 8 ensures that paired image and answer choices should be more similar than unpaired ones (i.e., $\text{cosine}(\mathbf{h}, \mathbf{s}_+) > \text{cosine}(\mathbf{h}, \mathbf{s}_-)$). $\mathbf{s}_+$ denotes the embedding of a paired annotation, $\mathbf{s}_-$ is a sampled statement embedding in the mini-batch, using semi-hard mining [218], and η is the margin in the triplet loss.

$$L(\mathbf{h}, \mathbf{s}) = \max(0, \text{cosine}(\mathbf{h}, \mathbf{s}_-) - \text{cosine}(\mathbf{h}, \mathbf{s}_+) + \eta) \tag{8}$$

In the sections below, we describe in detail how we represent the ad image $\mathbf{h}$ using a graph, which involves information from image regions, text in the image (if available), and potentially noisy retrieved external information. For the human-annotated action-reason statements, we use a Bi-directional Long Short-Term Memory (BiLSTM) model to encode them into the D-dimensional joint feature space $\mathbf{s} = \mathbf{W}_s \textsc{BiLSTM}(\psi(s); \boldsymbol{\theta}_s) \in \mathbb{R}^{D \times 1}$, where ψ is the word embedding process, $\boldsymbol{\theta}_s$ denotes the parameters of the statement encoder, and $\mathbf{W}_s$ is for the linear layer. During inference, models pick the most probable statement from candidates according to cosine similarity: $\underset{\mathbf{s} \in \text{candidates}}{\text{argmax}} \ \text{cosine}(\mathbf{h}, \mathbf{s})$.

5.2.3 Image Representation Graph: Nodes and Edges

Briefly, an image is partially represented using slogan text found in the image; in turn, these slogans are represented using external information found using the slogans as queries. Our image representation graph contains four types of nodes (image, slogan, knowledge and a global node), and three types of edges connecting these nodes.

Image nodes. For each image proposal $v_i \in V$, we use a pre-trained model to extract its feature $\text{CNN}(v_i)$. The embedding of v_i, denoted as $\mathbf{v}_i \in \mathbb{R}^{D \times 1}$, is obtained as a linear projection $\mathbf{v}_i = \mathbf{W}_v \text{CNN}(v_i)$ where $\mathbf{W}_v$ is the parameter.

Slogan nodes. We represent each OCR-detected textual slogan $t_i \in T$ using a BiLSTM encoder, then linearly project it into the same feature space as the image: $\mathbf{t}_i^{(0)} = \mathbf{W}_t \textsc{BiLSTM}(\psi(t_i); \boldsymbol{\theta}_t) \in \mathbb{R}^{D \times 1}$, where: ψ is the shared word embedding process, $\boldsymbol{\theta_t}$ and $\mathbf{W}_t$ denote the parameters. As OCR may produce noisy detections, model weights $\boldsymbol{\beta}$ discussed below (Eq. 10) choose which OCR results to use.

Knowledge nodes. Since the embedded slogans in ads are usually succinct, abbreviated, or ambiguous [128, 318], an external database will be used as a source of knowledge to help enriching and clarifying the meaning of the slogans. Specifically, we send each word in slogan t_i to the DBpedia knowledge base [134] as a query. This retrieval process φ

returns a set of related comments. For example, $\varphi(\text{“WWF”})$[1] returns the explanations of “Windows Workflow Foundation”, “Words with Friends”, “World Wide Fund for Nature”, and so on. We take the union of the retrieved knowledge entries to enrich a slogan, denoted as $\phi(t_i) = \bigcup_{q \in t_i} \varphi(q)$. In Fig. 17, the blue boxes show these extended pieces of knowledge for a specific slogan. The above procedure aims for high coverage rate, thus unavoidably many of the retrieved knowledge pieces will be irrelevant, but our model will learn to select the relevant ones, using the weights $\boldsymbol{\alpha}$ in Eq. 9.

For the external knowledge $k_{i,j} \in \phi(t_i)$ retrieved for the slogan (with j ranging over all retrieved comments for slogan t_i), we use a separate encoder $\mathbf{k}_{i,j} = \mathbf{W}_k \text{BiLSTM}(\psi(k_{i,j}); \boldsymbol{\theta}_k) \in \mathbb{R}^{D \times 1}$ with parameters $\boldsymbol{\theta}_k$ and $\mathbf{W}_k$, to encode the information. Note that knowledge nodes share the word embedding process ψ with slogan nodes and human-annotated statements but not the BiLSTM encoder, because we suppose word meanings in different modalities (DBpedia comments, slogans, action-reason statements) are the same, but the grammar structures may differ.

Edges. We build an inference graph (DAG) to capture the relationships for a better understanding of the image. We treat all the proposals, slogans, and knowledge pieces as nodes, with the knowledge nodes connected to the associated slogans by `IsADescriptionOf` edges. Next, we add a *global node* as an overall representation and connect all proposals and slogans to it using `ContributesTo` edges. The representation of the global node will be used to facilitate message passing and graph inference (described next). We also add extra `IsIdenticalTo` self-looping connections to all slogan nodes. Fig. 17 shows an example.

5.2.4 Image Representation Graph: Inference

Our method propagates information in a bottom-up manner and adjusts edge weights to optimize the final image representation $\mathbf{h}$ (Eq. 8). This inference procedure is similar to the Graph Convolutional Network (GCN) [120] in that we both use message passing to deduce the uncertain node embeddings. However, we fuse global context information to compute the edge weights, while GCN considers only the local information among neighbors.

[1] `http://dbpedia.org/page/WWF`

Updating slogan embeddings. The slogan representation will fuse messages from slogan and external knowledge. We define a weight vector $\boldsymbol{\alpha}_i \in \mathbb{R}^{1+|\phi(t_i)|}$ to denote the incoming edge scores for a slogan node t_i, where $\alpha_{i,0}$ is the weight of the self-loop edge `IsIdenticalTo`, and $\alpha_{i,j}$ $(j \in \{1, \ldots, |\phi(t_i)|\})$ are the weights of `IsADescriptionOf` edges. We require that $\sum_{j=0}^{|\phi(t_i)|} \alpha_{i,j} = 1$. Eq. 9 defines the policy for updating t_i. It requires the slogan t_i to choose a meaning (soft selection using the $\boldsymbol{\alpha}$ weights) among its initial embedding $\mathbf{t}_i^{(0)}$ and representations of the retrieved DBpedia comments $\mathbf{k}_{i,j}$. We describe how we learn $\boldsymbol{\alpha}$ weights shortly.

$$\mathbf{t}_i^{(1)} = \underbrace{\alpha_{i,0}\mathbf{t}_i^{(0)}}_{\text{original meaning}} + \underbrace{\sum_{j=1}^{|\phi(t_i)|} \alpha_{i,j}\mathbf{k}_{i,j}}_{\text{descriptions from extra knowledge}} \tag{9}$$

Computing global embedding. We infer the embedding of the global node from the direct information in image patches and slogans, and the indirect information of knowledge pieces. Specifically, we define a vector $\boldsymbol{\beta} \in \mathbb{R}^{|V|+|T|}$ denoting the weights of different `ContributesTo` edges. The first $|V|$ values are the contributions of image proposals and the next $|T|$ denote slogans. We require $\sum_{i=1}^{|V|+|T|} \beta_i = 1$. The global embedding $\mathbf{h}$ is a weighted sum of proposal and updated slogan embeddings.

$$\mathbf{h} = \underbrace{\sum_{i=1}^{|V|} \beta_i \mathbf{v}_i}_{\text{messages from proposals}} + \underbrace{\sum_{i=|V|+1}^{|V|+|T|} \beta_i \mathbf{t}_i^{(1)}}_{\text{messages from slogans}} \tag{10}$$

5.2.5 Image Representation Graph: Learning Edge Weights

We use an image-guided attention mechanism to infer $\boldsymbol{\alpha}$ (Eq. 9) hence choose whether to incorporate the external information or maintain the original slogan feature. This choice depends (1) the relation between the node and the connected slogan target, and (2) the relation between the node and the image context. We use a group of three-layer perception models denoted as $\text{MLP}(\mathbf{x}, \mathbf{y}; \boldsymbol{\theta})$ to model the relations between any two types of feature vectors $(\mathbf{x}, \mathbf{y} \in \mathbb{R}^{D\times 1})$. In Eq. 11, $[;]$ denotes concatenation, and $\cdot$ point-wise multiplication; $\boldsymbol{\theta} = (\mathbf{W}_1, \mathbf{W}_2)$ denotes parameters of a specific relation MLP, in which $\mathbf{W}_1$, $\mathbf{W}_2$ are parameters.

$$\text{MLP}(\mathbf{x}, \mathbf{y}; \boldsymbol{\theta}) = \mathbf{W}_2 \tanh(\mathbf{W}_1 \left[\mathbf{x}; \mathbf{y}; \mathbf{x} \cdot \mathbf{y}\right]) \tag{11}$$

Eq. 12 defines the edge weights connecting to textual slogans t_i. We define the image context $\bar{\mathbf{v}} = \frac{1}{|V|} \sum_{i=1}^{|V|} \mathbf{v}_i$. $\boldsymbol{\theta}^t_\alpha$ and $\boldsymbol{\theta}^c_\alpha$ are the parameters of the node-slogan and node-context MLPs. These MLPs measure how strong is the relationship between a node and the target slogan, and between a node and the image context.

$$\begin{aligned} a_{i,j} &= \begin{cases} \text{MLP}(\mathbf{t}_i^{(0)}, \mathbf{t}_i^{(0)}; \boldsymbol{\theta}^t_\alpha) + \text{MLP}(\mathbf{t}_i^{(0)}, \bar{\mathbf{v}}; \boldsymbol{\theta}^c_\alpha) & \text{when } j = 0 \\ \text{MLP}(\mathbf{k}_{i,j}, \mathbf{t}_i^{(0)}; \boldsymbol{\theta}^t_\alpha) + \text{MLP}(\mathbf{k}_{i,j}, \bar{\mathbf{v}}; \boldsymbol{\theta}^c_\alpha) & \text{when } 1 \le j \le |\phi(t_i)| \end{cases} \\ \boldsymbol{\alpha}_i &= \text{softmax}(\boldsymbol{a}_i) \end{aligned} \tag{12}$$

To compute weight vector $\boldsymbol{\beta}$, we update the slogan context $\bar{\mathbf{t}}^{(1)} = \frac{1}{|T|} \sum_{i=1}^{|T|} \mathbf{t}_i^{(1)}$, then use Eq. 13. This is a co-attention mechanism in that we use visual context to determine weights of slogan nodes, and use slogan context to decide contributions of image proposals. When there is no slogan detected, the image features will dominate.

$$\begin{aligned} b_i &= \begin{cases} \text{MLP}(\mathbf{v}_i, \bar{\mathbf{t}}^{(1)}; \boldsymbol{\theta}^v_\beta) & \text{when } 1 \le i \le |V| \\ \text{MLP}(\mathbf{t}_i^{(1)}, \bar{\mathbf{v}}; \boldsymbol{\theta}^t_\beta) & \text{when } \ |V| + 1 \le i \le |V| + |T| \end{cases} \\ \boldsymbol{\beta} &= \text{softmax}(\boldsymbol{b}) \end{aligned} \tag{13}$$

The weight vectors $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ allow our model to choose which knowledge pieces and slogans to use. We show the knowledge pieces chosen (with α larger than 0.05) in Fig. 18; thicker arrows correspond to larger values of α, β.

5.2.6 Masking to Break Shortcuts

As we show in our experiments, combining the knowledge directly with the image and text, despite the learned edge weights, achieves small gains over using image and text alone. As we show in Fig. 18, our model as described so far often ascribes small weights $\boldsymbol{\alpha}$ to external knowledge retrieved. We discussed this "shortcut learning" phenomenon in Sec. 5.1. Thus, we next focus the model's attention towards important cues and knowledge pieces for reasoning, using a set of automatic masking strategies. To cope with this problem, we propose a simple yet effective masking strategy to break shortcut learning. For example, we replace the query from the retrieved paragraph with the out-of-vocabulary token. In this way, the two pieces of knowledge in Fig. 17 become "`[oov]` is a sportswear company" and "`[oov]` is the name of an asteroid". Then the model can figure out whether "sportswear" or "asteroid" helps more for understanding the ad. At test time, when the model sees a rare sportswear company, it can benefit from the retrieved knowledge and not fail due to failed word-matching.

Our masking is similar to dropout (which we do use for our baseline), but applied over pieces of evidence in the slogan, knowledge comments, or action-reason statements. It is also similar to masking in cross-modal transformer methods [152, 37] but (1) we do not train the method to recover the masked symbol, and (2) transformer methods do not employ external knowledge, which is the key focus of our work.

We experiment with the following masking strategies:

- M_t randomly drops a detected textual (T) slogan, with a probability of 0.5.
- M_s randomly sets the query words (e.g. "WWF" or "Nike") in the human-annotated statements (S) to the out-of-vocabulary token, with probability 0.5.
- M_k replaces the DBpedia queries in the retrieved knowledge contents with the out-of-vocabulary token.

We found the masking strategy helps to significantly improve the main task of retrieving an answer. Moreover, when we evaluated the relevance of the knowledge pieces our model chose using weights $\boldsymbol{\alpha}$, we found an even more significant margin. While our masking strategy is specific to our target domain, masking in general merits exploration as a technique to aid in knowledge-based reasoning.

5.3 Experiments

5.3.1 Implementation and Experimental Setup

Dataset. We use the data from the 2018 ad understanding challenge[2]. There are 51,223 *trainval* images paired with 161,557 annotated statements; and 12,805 *test* images, each with 3 correct statements and 12 incorrect distractions (15 in total). We use Google Cloud Vision OCR[3] to recognize the embedded textual slogans. We retrieve DBpedia comments based on detected slogans; an example SPARQL query is shown in our supplementary file. Eventually we obtain 443,747 detected textual slogans, and 30,747 unique knowledge descriptions, to be associated with the 64,028 images (*trainval*+*test*). Each image is annotated with, on average, 6.9 slogans and 27.5 DBpedia comments.

We recruit human annotators to manually verify whether the retrieved knowledge is helpful for the ad understanding task. Specifically, for a given advertisement, we show all retrieved knowledge pieces and ask humans to annotate whether each piece is helpful or not in understanding the ad. These annotations serve as "gold standard" for knowledge selection evaluation (Sec. 5.3.4). We provide details in supp. We emphasize these annotations are never used to train.

Metrics and settings. Following the convention in the Ads challenge, we report accuracy (aka. precision@1) to compare against other methods from the challenge. However, we note that statement retrieval *accuracy* on the original task (3 correct with 12 incorrect statements) is not distinguishable enough, as many methods tie on this metric. To mitigate this issue, on one hand, we additionally report *rank* and *recall@K* scores inspired by [115, 121, 265]. For the rank metrics, we report min, and avg rank of the three correct statements. On the other hand, we created two additional "harder" test sets named Sampled-100 and Sampled-500, where each image is accompanied by 3 correct statements and 97 (or 497) incorrect

[2] `http://evalai.cloudcv.org/web/challenges/challenge-page/86`

[3] `https://cloud.google.com/vision/`

distracting options. We compare different models on these two challenging tasks as well as the original Ads challenge.

DBpedia knowledge. To increase recall, we add DBpedia entries that have *wikiPageDisambiguates* or *wikiPageRedirects* properties. This results in 17,277 anchored queries, associated with 30,747 DBpedia comments. We provide more details in the supplementary file.

Training details. We use a pre-trained object detector [299] to generate 10 proposals per image. We keep the 20 largest OCR detected regions. Our vocabulary for slogan, knowledge and statements consists of words that appeared more than 5 times in human-annotated statements or more than 20 times in OCR slogans or DBpedia comments. $\mathbf{v}_i$, $\mathbf{t}_i^{(0)}$, $\mathbf{t}_i^{(1)}$, $\mathbf{k}_{i,j}$, $\mathbf{h}$, $\mathbf{s}$ are all 200-D vectors. We use RMSprop with learning rate 0.001, batch size 128, and η (in triplet loss) of 0.2. More details are provided in supp.

5.3.2 Qualitative Examples

In Fig. 18, we show the graphs learned by our model. In general, with the masking mechanism we observe that the model focuses more on useful knowledge.

- The weights (width of arrow) from visual objects, slogans and external knowledge towards the global node (star) reveal their relative contributions.
- The model without masking does not utilize the external knowledge effectively: all knowledge pieces have extremely small weights thus are omitted from the visualization. This indicates that even though the external knowledge is available, the model still tends to process superficial word pattern matching. Instead, when the entity information (potential shortcut) is masked from the retrieved comments, along with other info randomly sampled and masked, the model learns semantics from the external knowledge.
- These results also show the importance of evaluating knowledge selection explicitly as a side task, as models may solve the main answering task but not use external knowledge (which should intuitively be helpful) at all.

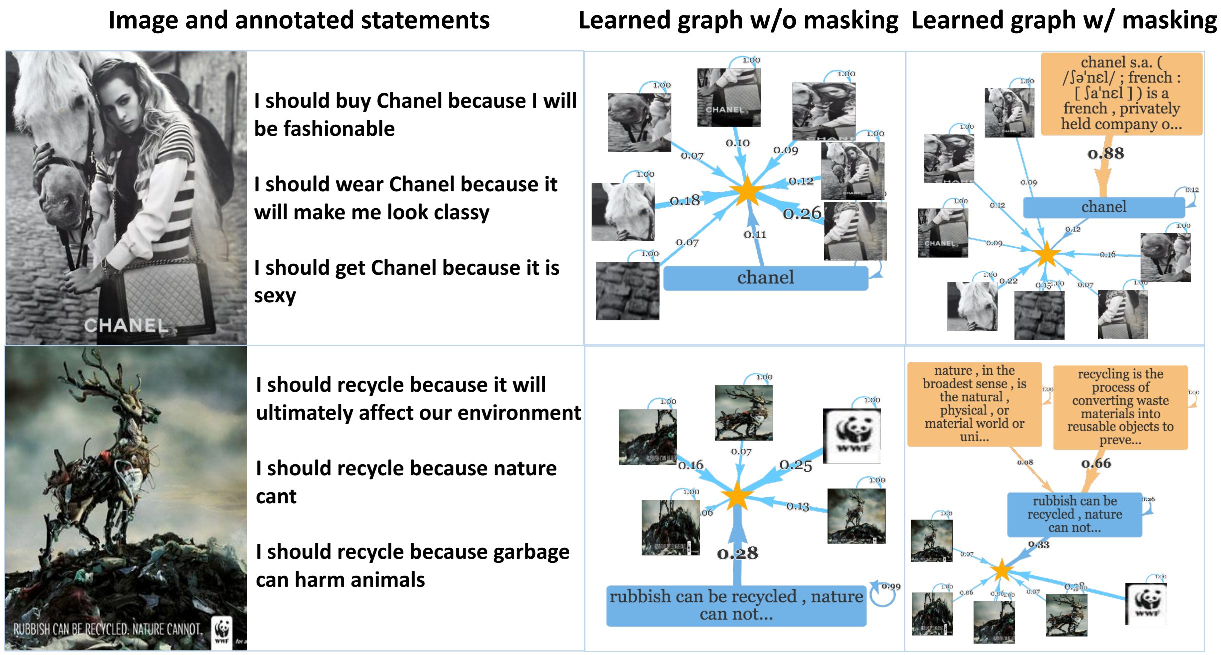


Image and annotated statements
Learned graph w/o masking
Learned graph w/ masking
I should buy Chanel because I will be fashionable
I should wear Chanel because it will make me look classy
I should get Chanel because it is sexy
CHANEL
chanel
0.10
0.09
0.07
0.12
0.18
0.26
0.07
0.11
chanel s.a. (/ʃəˈnɛl/ ; french : [ʃaˈnɛl]) is a french , privately held company o...
0.88
chanel
0.09
0.12
0.12
0.16
0.22
0.15
0.07
I should recycle because it will ultimately affect our environment
I should recycle because nature cant
I should recycle because garbage can harm animals
RUBBISH CAN BE RECYCLED. NATURE CANNOT.
WWF
0.07
0.16
0.25
0.13
0.28
rubbish can be recycled , nature can not...
0.99
nature , in the broadest sense , is the natural , physical , or material world or uni...
recycling is the process of converting waste materials into reusable objects to preve...
0.08
0.66
rubbish can be recycled , nature can not...
0.33
0.07

Figure 18: Examples of the learned graphs (best with zoom). We show the ad image and annotated action-reason statements on the left, the graph learned without masking in the middle, and that learned with masking (our approach) on the right. We show slogans in blue, DBpedia comments in orange, and the global node as a star. **Arrow thickness is correlated with learned weights α, β.** For visualization we removed all edges with small weights (threshold=0.05). Our method effectively leverages external information, as it relies on appropriate knowledge (in orange) more than the baseline method w/o masking.

5.3.3 Main Result: Effectiveness of Masking

In Tab. 16, let V denote the visual proposals, T the textual slogan information, and K the knowledge comments from DBPedia. M_t, M_s, and M_k, denote the different masking strategies described in Sec. 5.2.6. Simply "M" (for mask) means we use all three of them. The V,T method resembles [297], but uses a graph to represent the image and slogan. By comparing V,T and V,T+K in each task, we see that simply adding knowledge achieves very marginal gains because the benefit of knowledge gets drowned-out due to shortcuts. However, *our masking strategy* OURS: V,T+K(M) *improves results on all tasks and almost all metrics.* Accuracy (P@1) provides limited information because it only measures the easy-to-predict cases and all models are doing equally well. However, with the ranking metric and on the more challenging Sampled-100 and Sampled-500 test sets, we see our masking strategy brings significant and consistent performance gains. *Further, masking in conjunction with applying external knowledge (last row in each group) achieves better results compared to* not *using knowledge (first row).* Our method allows better reasoning (through external knowledge) by mitigating the effect of shallow matches (through masking).

Table 16: Main result using three ranking task setups. The best model in each group is shown in **bold**. High Precision and Recall scores, and low Rank scores, are better.

Method	P@1	P@3	P@5	P@10	R@1	R@3	R@5	R@10	Min Rank	Avg Rank
Results on the Challenge-15 task										
V,T	**87.3**	76.6	55.1	30.6	**28.4**	74.2	87.9	97.5	1.26	3.02
V,T+K	**87.3**	76.6	55.1	30.6	**28.4**	74.3	87.9	97.6	1.25	3.02
OURS: V,T+K(M)	**87.3**	**77.5**	**55.9**	**30.8**	**28.4**	**75.2**	**89.2**	**98.2**	**1.23**	**2.91**
Results on the Sampled-100 task										
V,T	79.8	66.5	46.9	26.2	26.0	64.4	74.9	83.5	2.38	7.52
V,T+K	80.0	67.0	47.0	26.1	26.0	64.9	75.1	83.4	2.29	7.49
OURS: V,T+K(M)	**80.2**	**67.9**	**47.9**	**26.8**	**26.1**	**65.8**	**76.6**	**85.4**	**2.14**	**6.56**
Results on the Sampled-500 task										
V,T	**65.5**	52.3	37.8	21.7	**21.3**	50.5	60.4	69.0	8.18	30.1
V,T+K	65.4	52.3	38.0	21.9	**21.3**	50.6	60.7	69.6	7.60	30.0
OURS: V,T+K(M)	64.8	**52.4**	**38.3**	**22.1**	21.1	**50.7**	**61.1**	**70.6**	**6.89**	**25.1**

Tab. 17 shows an ablation study using the average rank. The table includes results for all three tasks, and we use the evaluation on the most difficult Sampled-500 to describe our improvement percentages. First, directly adding knowledge (V,T+K v.s. V,T) does not help. The +K leads to only 0.5% improvement which is negligible (29.96 v.s. 30.11).

Table 17: Average Rank on the ranking tasks. Relative improvement is based on Sampled-500. Lower scores are better. The best method is shown in **bold**.

Method	Challenge-15	Sampled-100	Sampled-500	Relative to V,T+K
V,T	3.02	7.52	30.11	
V,T+K	3.02	7.49	29.96	
V,T+K(M_t,M_s)	2.97	7.05	27.66	+7.68%
V,T+K(M_t,M_k)	2.93	6.74	26.04	+13.08%
V,T+K(M_s,M_k)	3.00	7.43	29.64	+1.07%
V,T+K(M_t,M_s,M_k)	**2.91**	**6.56**	**25.14**	+16.09%

However, if we apply masking to mitigate the effects of shortcut learning, the performance is improved by a large margin. As we compare V,T+K(M_t,M_s,M_k) to V,T+K, the average rank is reduced from 29.96 to 25.14 (-4.82 average rank or *+16.09% relative improvement when we use our proposed masking*). Further, we verify that removing any of the masking mechanisms (V,T+K(M_t,M_s), V,T+K(M_t,M_k), and V,T+K(M_s,M_k)) leads to inferior performance (27.66, 26.04, 29.64 v.s. 25.14). We conclude the useful information of the external knowledge is fully unleashed if and only if shortcut learning can be suppressed.

5.3.4 Side Task: Analyzing the Knowledge Utilization

We check whether the methods know the usefulness of the potentially irrelevant retrieved knowledge for ad understanding. We use a side task which requires no additional training, to test their knowledge selection and filtering power. Specifically, we use the edge weights methods learned, with and without our masking strategy. Note that methods did *not* receive supervision for this task at training time; instead, our masking strategy helps our method accomplish the task better than the baseline can. To the best of our knowledge, similar experiments have not been done in prior visual reasoning work. Even for the latest KBQA datasets, all provided knowledge pieces are relevant information without noise. However, in our setting, the retrieved DBpedia knowledge pieces are usually noisy, while only a few of them could expand the image's meaning. Such noisy retrieval is more likely to happen in real-world applications.

We measure how accurately the model could select the useful knowledge pieces from the noisy candidate pool. For each image, we take the learned weights for DBpedia comments (Eq. 9) as a knowledge importance score, and select the one with highest score using $\text{argmax}_{i,j}\, \alpha_{i,j}$. Then the model-selected knowledge is compared against human annotations, for an accuracy score. The procedure is *integral to the main task because the weights are learned automatically in it.* The results are shown in Tab. 18. V,T+K(M_t,M_s,M_k)'s improves accuracy to 54.4% compared to 25.2% for V,T+K (+115% improvement!) *In other words, masking doubles the ability of our method to retrieve appropriate external knowledge, by removing reliance on shortcuts.* Further, this result shows the discrepancy of the main and side metrics (16% gain for our method in Table 17 compared to 115% in Table 18).

Table 18: Accuracy(%) on the knowledge selection task

Methods	Accuracy (%)
V,T+K	25.2
V,T+K(M_t,M_s)	**54.4**
V,T+K(M_t,M_k)	53.0
V,T+K(M_s,M_k)	25.9
V,T+K(M_t,M_s,M_k)	52.6

5.3.5 Comparison with the State-of-the-art

We compare our model to the approaches in the "Automatic Understanding of Visual Advertisements" challenge and some latest works. Vse trained by [294] uses only the image-level feature to represent the ad and triplet loss to optimize the model. AdNet [86] is similar but uses ResNet as the network backbone. Advise [294] aggregates proposal feature vectors to get the image representation. It incorporates knowledge from a pre-trained dense captioning model [107] and a symbol classifier. CyberAgent [195] is the first model that uses slogan texts embedded in the image. Rhetoric [297] is a hybrid model of both Advise and CyberAgent; it uses pointwise addition to integrate image and slogan, and is the current state-of-the-art.

Table 19: Accuracy(%) on the ads-challenge. We compared our method to state-of-the-art models, using the data split provided in the 2018 ads-challenge

Methods	Accuracy (%)
Vse [294]	62.0
AdNet [86]	65.0
Advise [294]	69.0
CyberAgent [195]	82.0
Rhetoric [297]	83.3
Ours	**87.3**

Tab. 19 shows the comparison to these approaches. Our model outperforms even the strongest baseline Rhetoric by 4.8% in terms of accuracy (87.3% v.s. 83.3%). While Rhetoric also incorporates both image and slogan information, our method represents this information in a more fine-grained manner using the graph. Besides, our method uses external knowledge from DBPedia.

5.4 Conclusion

Visual reasoning has attracted much attention, although the "reasoning" process is usually hidden behind a mixed or decoupled evaluation protocol. One of the main contributions of this chapter is that we proposed a side task in addition to the ranking of the ads action-reason pairs - choosing the correct knowledge piece. In the side task, models do no rely on supervised learning to select knowledge. Instead, they determine based on if the knowledge is helpful to understand the ad. In other words, knowledge choosing is learned through weak supervision.

For the thesis topic, this chapter proved the hypotheses H1, H2 (see Tab. 20). This chapter provides an efficient way to use embedded slogans and external DBpedia comments - through both a bottom-up graph model and a stochastic masking technique. The graph model determines the detected noisy slogans and the retrieved unrelated DBpedia descriptions, then down-weigh them by decreasing the graph edge scores. The masking strategy forces the model to make predictions based on its reasoning over external knowledge pieces and avoids using unreliable evidence. Our model achieved state-of-the-art performance on the challenging ads understanding task. It achieved a 5% improvement in terms of accuracy as compared to Chapter 3.

Table 20: Conclusion - validated hypotheses in this chapter.

	Multimodal features help to understand images/videos with implicit persuasive intent, such as visual advertisements.	Text features can be unreliable if not modeled appropriately.	Text supervision contains noise, but can be used to localize visual objects in space, if modeled properly.	Text supervision provides contexts regarding visual objects, they are reliable cues for disambiguating entities and relations.	Noisy audio narrations as a multimodal signal can be modeled to localize video actions in temporal domain.
This Chapter	✓	✓			

6.0 A Case Study of the Shortcut Effects in Visual Commonsense Reasoning

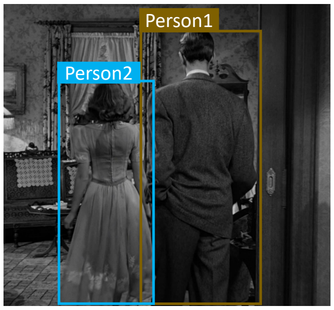
6.1 Introduction

We explored methods to use multimodal features and external knowledge efficiently in the last chapter. In this chapter, we focus on a more comprehensive visual reasoning task (the Visual Commonsense Reasoning) and want to see if unreliable evidence broadly exists. Specifically, we seek if "shortcuts" exist in another form that also connects inputs and outputs, but provides misleading cues. Different from the previous chapter: (1) we quantify the shortcut effects by constructing adversarial datasets, in which more drop of performance of existing models means more severe consequences; (2) we do not assume the noisy multimodal inputs scenario; (3) our solution uses the end-to-end-trained transformer models without mimicking human reasoning.

Models for vision-and-language (VL) tasks, such as visual question answering (VQA) and visual commonsense reasoning (VCR), perceive the features of an image and provide natural language responses regarding the visual contents. The comprehensiveness of the VQA process seems to require complete human-like intelligence, and has inspired great interest. Unfortunately, in practice, models have many opportunities to bypass "reasoning" and instead find shallow patterns in the data in order to match answers to image-question pairs. By "reasoning" we mean a generalizable process that analyzes the structure of the world as demonstrated by training data, pays attention to links between participants in the scene as well as between entities and their semantic properties, and analyzes how these correspond to the entities or events indicated in the question. Such a process ideally persists when small changes are made to the potential answer options without changing their meaning, because the entities represented by these options remain the same.

To solve visual question answering tasks, studies used feature fusion [12, 326], attention [154, 62, 7], or question-related modular structures [9, 8]. Recently, transformers modeling cross-modal attention [5, 37, 152, 140, 240] have also been applied. These methods are trained with supervision i.e. the correct answers are provided by human annotators on the training set. The nature of supervised training means methods are rewarded for finding *any* connection between inputs (image-question) and outputs (answer options). In other words, methods can do well without performing complex reasoning, if they can find enough “shallow” matches between input and output. We refer to such shallow matches as “shortcuts”.

Shortcut effects: Example and definition. Consider the example in Fig. 19 from the VCR dataset [314]. In the figure, [person1] (male) is on the right and [person2] (female) is on the left. The correct option has the most overlap with the question: the “[person1]” and “[person2]” tags, and the word “dress”. Thus, to answer this question, the model need not perform reasoning or even look at the image. Examples in VCR vary: not all contain shortcuts of this nature, yet others contain even more severe shortcuts. For example, some incorrect answer choices mention entities entirely unrelated to question and image, which are thus easy to eliminate.



Question: What does ***[person1]*** *think of* ***[person2]****'s* ***dress****?*

Correct answer: ***[person1]*** *thinks* ***[person2]*** *looks stunning in her* ***dress****.*

Incorrect #1: She does not approve.

Incorrect #2: [person2] is a girl and girls like to wear makeup.

Incorrect #3: [person1] is confused and annoyed by [person2] following her in the store.

Figure 19: Shortcut effects: An example.

We define “shortcuts” as a way of achieving the correct answer by simply matching repeated references to the same entities in the question and answer options. We find that in 67.8% samples for the Q→A task in VCR, and 65.2% samples for the QA→R task, the correct choices have the most overlapped referring tags among the candidates. Further, state of the art methods’ performance drops significantly when these shortcuts are removed.

One reason for shortcuts is that humans often repeat the keywords or essential entities of the question to give a complete answer; this is hard to avoid during data collection. Further, the shortcuts may have broader forms across different modalities. E.g., in language “excited” is a common association to “feeling”, people often perform action “eating” at visual environment “restaurant”, etc. We emphasize that researchers that train models for VCR should pay more attention given these inevitable shortcuts. Yet, prior methods have sometimes exacerbated shortcuts. E.g., the “grounding” of objects in [314] enables feature-level shortcuts since the same object feature may appear in both question and answer. We specifically examine shortcuts in the case of VCR, while the same phenomenon is likely to present in other datasets where question-answering is formulated as multiple-choice task and features full-sentence answers e.g. [249, 311].

While machine learning methods for other tasks also find easy ways to do well at the target task, we argue that “shortcuts” are a particular type of dataset bias whose reduction requires specific mechanisms. What exacerbates the problem is that such shortcutting is easier in the multiple-choice VQA setting compared to classification. In image classification, a shortcut has to be found across modalities, i.e. pixels to labels. In VQA, a shortcut between input and output can easily be found within the same modality, i.e. text in the question and text in answers. However, shortcuts are distinct from prior biases discovered in VQA datasets [77], because they have more to do with shallow string matches than modes in the answer distribution. No prior dataset bias work has studied shortcut effects.

In this work, we first quantify the impact of shortcuts on state-of-the-art models. We propose two methods to augment VCR evaluation. One makes small word-level changes while maintaining the original meaning, while the other examines which word a VCR method most depends on. We show the performance of SOTA methods drops significantly on the modified evaluation data. Second, we propose a novel masking technique to make training more robust and make models rely on more extensive evidence compared to individual shortcuts. Because masking may under-utilize useful information, we perform masking on curriculum, with a large masking ratio initially and gradually reducing it. We show our robustly trained method collapses less when partial evidence is missing, and curriculum masking is more effective than prior masking techniques in both the original and modified settings. Our paper is an initial exploration of shortcut effects in VQA and a case study of VCR. We expect it to inspire future ideas of overcoming shortcut effects. Our code and data are available at `https://github.com/yekeren/VCR-shortcut-effects-study`.

To summarize, our contributions are as follows:

- We show how to highlight the unreliable evidence in VCR to fool models relying on them.
- We show how to quantify the impact of shortcuts on different state-of-the-art models.
- We explored existing and novel masking techniques to make training more robust.
- We show that our robustly trained method collapses less when partial evidence is missing.

6.2 Approach

First, we develop techniques to quantify the detrimental effect of shortcuts, by removing some them at test time. Second, we propose a technique to make training more robust.

6.2.1 VCR Task and Basic Model

The visual Commonsense Reasoning (VCR) task involves two subtasks. The first one (Q→A) requires predicting if an answer choice $\boldsymbol{a}$ fits the context of both visual information $\boldsymbol{v}$ and question $\boldsymbol{q}$ (i.e., multi-choice VQA). The second subtask (QA→R) predicts the likelihood of a rationale $\boldsymbol{r}$, given $\boldsymbol{v}$, $\boldsymbol{q}$, and $\boldsymbol{a^*}$ ($\boldsymbol{a}^*$ is the correct answer). For each question, the dataset provides one correct choice (answer or rationale) as well as three distracting (incorrect) options. The evaluation protocol also involves a combined Q→AR metric without separate training. Fig. 21 shows examples of Q→A. Unlike other VQA datasets, VCR mixes person/object tag annotations with the questions and answers, denoting that the text refers to a particular image region. We find these tags create problematic shortcuts.

To achieve unified modeling $\mathcal{P}$ of both subtasks, we follow [5, 314, 308, 144] to reparameterize the formulation of QA→R (Eq. 14). We concatenate $\boldsymbol{q}$ and $\boldsymbol{a}^*$ to obtain question $\boldsymbol{q'}$ in QA→R, and treat rationale $\boldsymbol{r}$ as answer $\boldsymbol{a'}$. Thus both VCR models differ only in parameters $\boldsymbol{\theta}$, $\boldsymbol{\theta'}$.

$$\begin{aligned} Q \rightarrow A &: \mathcal{P}(\boldsymbol{v}, \boldsymbol{q}, \boldsymbol{a}; \boldsymbol{\theta}) \\ QA \rightarrow R &: \mathcal{P}(\boldsymbol{v}, \boldsymbol{q}', \boldsymbol{a}'; \boldsymbol{\theta}'), \text{where } \boldsymbol{q'}\!=\left[\boldsymbol{q}; \boldsymbol{a}^*\right],\ \boldsymbol{a'}\!=\!\boldsymbol{r} \end{aligned} \tag{14}$$

For our modified, more challenging evaluation setting (Methods to Evaluate the Shortcut Effects), we use four recent, diverse methods. To show improvements through robust training, we focus on B2T2 [5] to implement $\mathcal{P}$. We choose B2T2 because: (1) The architecture is simple. It is essentially a BERT [50] model with multimodal inputs, with the next sentence prediction of BERT modified to be the matching prediction of the answer given question-image pair. (2) BERT-based architectures are popular for the VCR task [5, 37, 152, 240, 138, 65, 306] hence our choice of method is representative. (3) B2T2 achieves good results without expensive pre-training on external, non-VCR data, while models like UNITER [37] are more dependent on expensive out-of-domain pre-training.

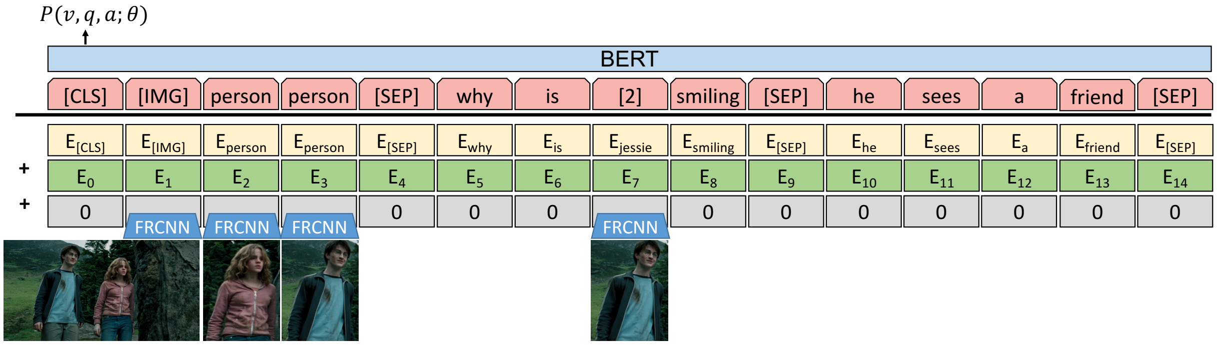


Figure 20: Model architecture for our shortcut effects study. We use BERT as the language model backbone and add the tag sequence features generated by Fast-RCNN to the token and positional embeddings. The contextualized feature of [CLS] is used to predict the answer-question matching score.

Fig. 20 shows how we predict the joint probability of $\boldsymbol{v}$, $\boldsymbol{q}$, $\boldsymbol{a}$. Similar to B2T2, we create a token sequence by concatenating the image object labels (from the VCR dataset, e.g. “person”) and textual words. We also create a tag features sequence using the associated Fast-RCNN features [69], adapted to the same dimensions as the word embeddings; for words not mentioning any visual objects, we pad with zeros. Then, the embeddings of the token sequence and the tag features are pointwise added and normalized before being fed to the BERT model to get the contextualized feature vectors. Next, we add a linear layer on the feature of the `[CLS]` token to estimate $\mathcal{P}(\boldsymbol{v}, \boldsymbol{q}, \boldsymbol{a}; \boldsymbol{\theta})$ (a scalar). We use *sigmoid cross-entropy* to optimize the model. Thus, $\mathcal{P}(\boldsymbol{v}, \boldsymbol{q}, \boldsymbol{a}; \boldsymbol{\theta})$ approximates a probability which is large if the answer is appropriate for this image and question. All models that we train, including baselines, use BERT-Base (12 layers, 768 hidden units) and ResNet-101 [82] pre-trained on ImageNet [47], as the language and vision models' backbones, respectively. We keep all the layers in BERT-Base trainable while we freeze the ResNet-101 layers until the ROIAlign. We use 4 GTX1080 GPUs, batch size of 48 (12 images per GPU), learning rate of 1e-5, ADAM optimizer, and the Tensorflow framework. We train for 50k steps (roughly 11 epochs) on the 212,923 training examples and save the model performing best on the validation set (26,534 samples), for each method in Table 25. Each model took 10 hours to train.

6.2.2 Methods to Evaluate the Shortcut Effects

We propose two methods (rule-based and adversarial) to modify the answer candidate options in the evaluation set. Both methods keep meanings unchanged in most cases, but the second does change meaning in some cases and is primarily used to gauge what kind of words in the answer options a VCR method relies on. The methods highlight shortcuts and test the models' capability of utilizing comprehensive features instead of shortcuts.

Rule-based modification. Inspired by the observations in Introduction, we first use a set of simple rules to modify references to persons. While individual words in the answers are changed, the meaning of the answer choices remains unchanged or almost unchanged. We always modify both the distracting and correct options. Depending on whether the question contains one or multiple person tags, we refer to the rule as RULE-SINGULAR or RULE-PLURAL. This method only covers a proportion of the validation data but causes a significant drop for several recent methods.

For ground-truth options, we turn person tags into pronouns to make the answer less associated with the question-image pair at the surface (removing tag matches). To choose the proper gender pronouns, we first check the hints ("his", "her", etc.) in both the question and answer. For groups of tags ("`[person1,person2]`"), we replace with the pronoun "they". Since the distracting options are semantically unrelated to the image, we assume the pronouns and person tags do not matter in *most* cases. We turn pronouns ("he", "she", "they") and any other person tags, into the person tags asked in the question. Tab. 21 shows some examples, where the question is about `[person2]`, and both "he" and `[person1]` are changed to `[person2]`.

Table 21: Examples - Modifying distractor answer options.

Question	Original	Changed to
Why is [person2] in such a rush?	*He used the wrong ingredients to make the meal.*	*[person2] used the wrong ingredients to make the meal.*
How is [person2] feeling?	*[person1] is very excited.*	*[person2] is very excited.*

Discussion: Shortcuts vs distribution shifts. Changing the distribution of the evaluation set compared to the training set naturally causes a drop in performance. What this modified evaluation allows us to do is measure precisely how much different methods rely on person tag shortcuts. Further, it creates a more realistic, less inflated setting to demonstrate the reasoning capacity of different models, including ours which enables robust training. The shortcuts we highlight through our modified evaluation, are distinct from distribution shifts. In particular, our robust training algorithm that copes with shortcuts (next section) improves performance in both the modified evaluation and the original setting. In contrast, a method that exploits the distribution shifts created with our modification by training on such modified data, degrades performance in the original setting.

Adversarial modification. We next propose an adversarial modification. First, we train a B2T2 model $\mathcal{P}(\boldsymbol{v}, \boldsymbol{q}, \boldsymbol{a}; \boldsymbol{\theta})$ to solve the VCR problem using unmodified data. Given ground-truth label information $\mathcal{C}(\boldsymbol{v}, \boldsymbol{q}, \boldsymbol{a}) \in \{0, 1\}$ ($\boldsymbol{a}$ *is* or *is not* the answer to $\{\boldsymbol{v}, \boldsymbol{q}\}$), we define the potential shortcut evidence in Eq. 15, where $|\cdot|$ denotes the length of the sequence and $\Psi(\boldsymbol{x}, i)$ is a function to replace the i-th token in sequence $\boldsymbol{x}$ with a special token `[MASK]`. Eq. 15 looks for the evidence in the answer choices that makes the model most "fragile", i.e. the special position in answer $\boldsymbol{a}$ such that after replacing that token with a mask, the cross-entropy loss is *maximized* (because we want to confuse models).

$$\begin{aligned}\mathbf{argmax}_{i\in[1,|\boldsymbol{a}|)]}[-&\mathcal{C}(\boldsymbol{v},\boldsymbol{q},\boldsymbol{a})\log\mathcal{P}(\boldsymbol{v},\boldsymbol{q},\Psi(\boldsymbol{a},i);\boldsymbol{\theta})\\ &-(1-\mathcal{C}(\boldsymbol{v},\boldsymbol{q},\boldsymbol{a}))\log(1-\mathcal{P}(\boldsymbol{v},\boldsymbol{q},\Psi(\boldsymbol{a},i);\boldsymbol{\theta})]\end{aligned} \tag{15}$$

Intuitively, there should be more than one word in the correct answer ($\mathcal{C}(\boldsymbol{v},\boldsymbol{q},\boldsymbol{a}) = 1$) that allows a method to find that answer. However, compared to the rule-based revisions, we expect that performance will drop for the adversarial setting because the adversarial method potentially changes the meaning. Thus, in this setting, we are more interested in what words cause performance to drop the most when masked, rather than how much performance drops. We provide statistics regarding the masked words in Experiments. Adversarial modification mostly attacks word repetitions, pronouns, and word tenses. This supports our intuition about shortcut effects: models use trivial, content-free hints to make decisions instead of real reasoning. We expect the rule-based modification to more precisely show the effect of a specific type of shortcut (person tag), while adversarial revision will show the broader effects in a less controlled environment (as any word can be chosen in Eq.15).

6.2.3 Robust Training with Curriculum Masking

We propose a new way to make training more robust such that it can overcome shortcut effects, using masking on a curriculum. We describe two masking baselines, then our new masking technique. Note the strategies we used to create the modified evaluation sets are not appropriate to augment the training set because they potentially add new shortcuts, as we show in experiments.

Masking baselines: Masked VCR and language modeling. We randomly replace tokens in answers with the `[MASK]` during training, with a probability of 5%, 10%, 15%, or 30%. We predict whether a masked answer follows the question, and refer to this technique as Masking in Experiments. The `[MASK]` token is not applied in inference. We also use masked language modeling (MLM), where the task is to predict the missing tokens in the masked sentence. We use a 0.001 coefficient to weigh the MLM softmax-cross-entropy loss; this is because too large weighting negatively affects the main loss (answer choice cross-entropy).

We jointly train for the two objectives and refer to the approach as Masking+MLM. Both of these masking strategies are inspired by BERT [50].

Our method: Masked VCR on a curriculum. There is a tradeoff between masking to increase robustness and maintaining the required information. We found that the more masking is applied during training, the better the result in the modified settings, but the worse it is on the original standard validation. Thus, we propose a new curriculum masking approach which slowly decays the amount of masking that is applied during training. It uses a high masking probability at the beginning, then gradually reduces the masking ratio:

$$\text{Masking ratio} = \text{Initial ratio} * e^{-(\text{Decay rate}*\text{Train steps})}$$

We feed hard examples (higher masking ratio) at the start because this regularizes the model to pay more attention to the inputs as a whole, while in later stages the model leverages examples that have closer distribution to the unmasked validation data. We refer to this method as Ours-CL, and show its benefit in Experiments. While curriculum learning [102, 312, 320, 103] has been tried to decide the *order* of tasks for pre-training [158, 267, 41], to our knowledge, ours is the first method to mask using a curriculum.

Discussion. None of our robust training approaches focus on pre-training on large external corpora, because its effect makes it unclear how a method makes its decisions, and this pre-training incurs a large computational cost. The contribution of pre-training on an external dataset gives mixed results: B2T2 [5] show pretraining on Conceptual Captions improves accuracy by 0.4%, vs 1% (and 2% for second-stage in-domain pre-training) for UNITER [37]. Our experiments with existing masking techniques resemble in-domain pretraining, but we show these are inferior to masking using a curriculum, in both the original and modified evaluation settings.

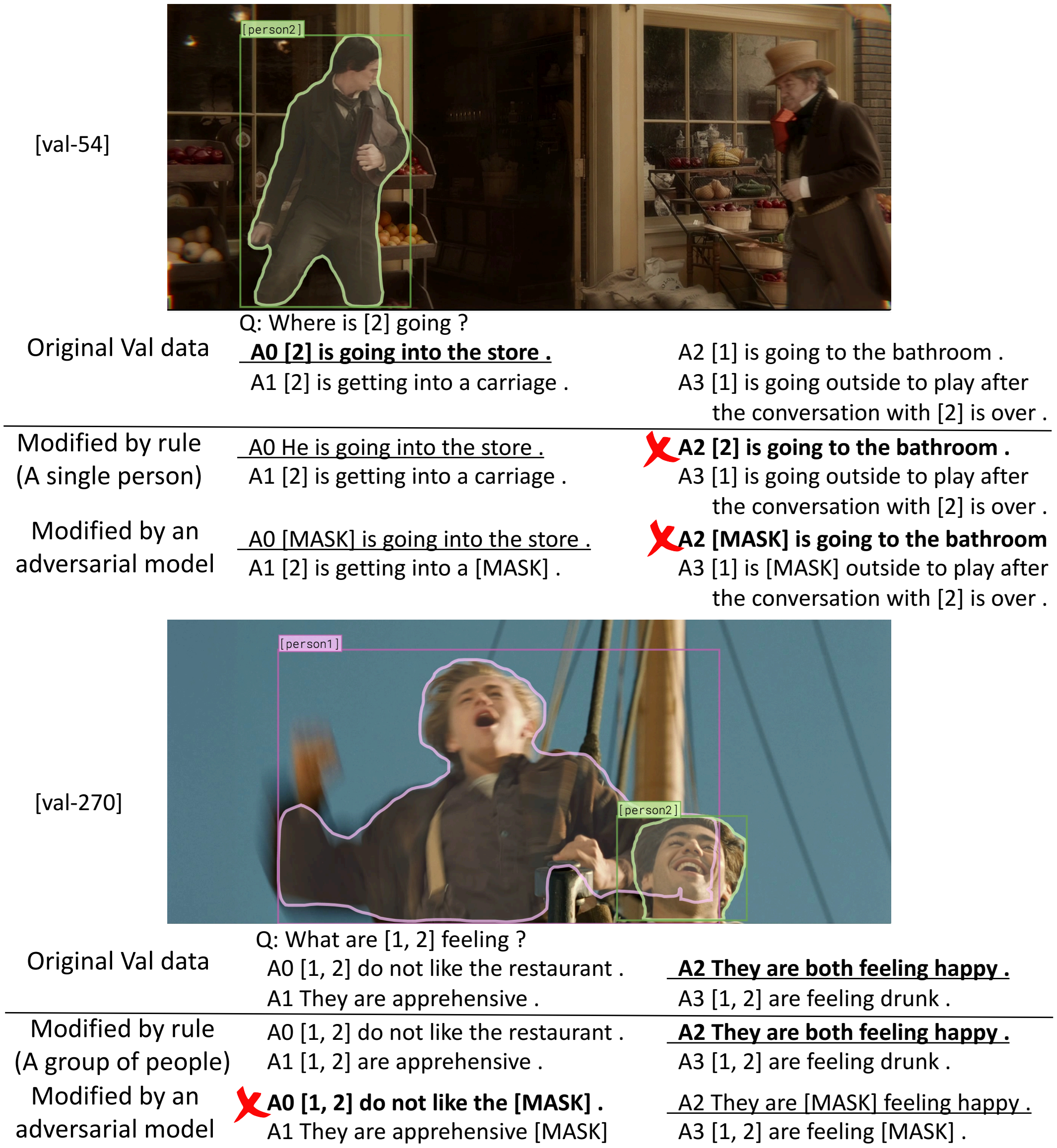
6.3 Experiments

We qualitatively demonstrate then quantitatively measure the effect of shortcuts through our modified evaluations, on four recent and competitive VCR methods. We then test how well our robust training strategy copes with the challenge.

6.3.1 Qualitative Results on the Modified Options

We show that R2C [314] (checkpoint by authors) is confused once the expected shortcuts are no longer available. In Fig. 21, we show the option chosen by the method in bold, and the correct one is underlined. In Fig. 21 top, in the original setting, only options A0 and A1 contain the person tag `[2]`, hence the model only had to rule out "carriage". In the rule-modified setting, the model confused "store" with "bathroom" once the easy way of ruling out non-matching references (`[person1]` v.s. `[person2]`) is no longer applicable. The adversarial method has detected the same shortcut, replaced `[person2]` with `[MASK]` and tricked the model. In Fig. 21 bottom, the model relied too much on detecting the incompatibility between the image and concept "restaurant": when the word "restaurant" is masked in the adversarial setting, the model chooses the incorrect option A0 rather than detecting the "happy" people.

The rule-based method, targeting the over-relying of person tags, is focused and precise. In comparison, it is not that intuitive what the adversarial method attacks. We hence show statistics of the top-20 masked words. In Tab. 22, p(mask x) denotes the frequency the adversarial method chose the token x to mask; $\sum_{\text{x}}$ p(mask x)=1. Since token appearance frequency varies, we also report p(mask x|exist x). We observe that the adversarial method chose to hide the top-20 words in most cases ($\sum_{\text{x}\in\text{top-20}}$ p(mask x)=45.37%). However, it is hard to say these words are crucial for human reasoning. For example, "#PERSON", "he", "they", "she" are pronouns referring to persons; "is", "a", "are" are articles with hints regarding numbers; "will", "going" involve tense information. "Not" and "yes" are two exceptions, and hiding them will change the meaning. However, the proposed adversarial method relies on no human intervention and such simple cases can be ruled out by extra



Original Val data

Q: Where is [2] going ?

A0 [2] is going into the store .
A1 [2] is getting into a carriage .
A2 [1] is going to the bathroom .
A3 [1] is going outside to play after the conversation with [2] is over .

Modified by rule (A single person)

A0 He is going into the store .
A1 [2] is getting into a carriage .
A2 [2] is going to the bathroom .
A3 [1] is going outside to play after the conversation with [2] is over .

Modified by an adversarial model

A0 [MASK] is going into the store .
A1 [2] is getting into a [MASK] .
A2 [MASK] is going to the bathroom
A3 [1] is [MASK] outside to play after the conversation with [2] is over .

Original Val data

Q: What are [1, 2] feeling ?

A0 [1, 2] do not like the restaurant .
A1 They are apprehensive .
A2 They are both feeling happy .
A3 [1, 2] are feeling drunk .

Modified by rule (A group of people)

A0 [1, 2] do not like the restaurant .
A1 [1, 2] are apprehensive .
A2 They are both feeling happy .
A3 [1, 2] are feeling drunk .

Modified by an adversarial model

A0 [1, 2] do not like the [MASK] .
A1 They are apprehensive [MASK]
A2 They are [MASK] feeling happy .
A3 [1, 2] are feeling [MASK] .

Figure 21: Qualitative study of shortcuts. We underline the ground-truth and bold the prediction of R2C. R2C was fooled by negligible changes in the answer options.

rules. Besides, they only constitute 2% of the revised evaluation data while the person tags are the leading choice of masking.

Many words in Table 22 are "content-free", in the sense that other nouns and verbs should intuitively be more important. We conclude that meaning does not change greatly when a single, however important, word is removed, yet method performance drops by 14-34%. We thus emphasize that researchers should pay special attention to the issue at both

Table 22: Statistics of top-20 words removed by the adversarial revision. Note how often content-free words (e.g. pronouns) are key for answering, hence removed.

Token x	p(mask x)	p(mask x| exist x)	Token x	p(mask x)	p(mask x| exist x)
#PERSON	25.71%	27.84%	will	0.77%	11.33%
.	3.82%	3.79%	to	0.65%	2.04%
he	2.53%	12.09%	going	0.59%	14.13%
is	1.56%	2.78%	are	0.59%	3.72%
they	1.54%	11.70%	feeling	0.56%	22.25%
not	1.29%	24.36%	him	0.47%	12.09%
she	1.20%	12.86%	it	0.41%	7.27%
yes	0.86%	22.47%	her	0.40%	8.99%
the	0.82%	2.97%	something	0.40%	11.62%
a	0.80%	3.06%	someone	0.39%	15.43%

the data acquisition and model learning phases. Besides VCR, shortcuts may also arise in other multiple-choice VQA tasks, e.g. MovieQA [249] and Social-IQ [311], when fragments of the question and answer can be trivially matched.

6.3.2 Shortcut Effects on Rule-based Modified Setting

We next quantitatively demonstrate how our modified evaluation setting affects the following four VCR methods.

- B2T2 [5] proposes early integration of visual features in BERT to benefit from stacked attention
- HGL [308] uses vision-to-answer and question-to-answer graphs using BERT/CNN embeddings
- TAB-VCR [144] incorporates objects and attributes into the R2C tag matching

- R2C [314] builds RNN layers on the pre-extracted BERT embeddings and uses attention mechanisms to highlight important visual/language elements

For HGL, TAB-VCR and R2C, we download the best-trained checkpoints provided by the authors and run inference using our modified validation. We refer to the reference implementation to implement B2T2, since no checkpoint was provided. Note B2T2, HGL and TAB-VCR are competitive in the VCR leaderboard, achieving ranks 17, 20, and 24. The better ranks are occupied by other BERT-based models [37, 152, 240, 138, 306, 65] focusing on pre-training using large external VL datasets and even object, attribute and relationship predictors [306]. These settings incur significant additional data collection cost.

Table 23: Shortcuts in VCR: rule-based modified evaluation.

Questions regarding	Count	Method	Q→A		QA→R	
			Std Val	Mod Val	Std Val	Mod Val
A single person e.g., *Where is [2] going ?* (Rule-Singular)	16,154	R2C	64.5	58.5	67.8	62.0
		HGL	69.8	66.1	70.8	64.5
		TAB-VCR	70.5	65.4	72.4	66.3
		B2T2	69.9	63.3	69.1	64.9
A group of people e.g., *What are [1,2] feeling ?* (Rule-Plural)	3,657	R2C	62.2	59.7	66.9	65.4
		HGL	69.2	67.5	70.7	69.8
		TAB-VCR	69.8	66.8	71.3	70.9
		B2T2	67.6	65.3	69.3	67.9

We observe that merely replacing the pronouns and person tags confuses the state-of-the-art models. Tab 23 shows the results. For Rule-Singular, the average drop in accuracy, between the standard and modified validation sets, is 5% for Q→A and 6% for QA→R. Although the performance of QA→R is better than that of Q→A in the original setting, the performance drop was higher on QA→R. Thus, we question if models have learned to reason instead of utilizing the shortcuts. The average drops for Rule-Plural are 2% and 1%, respectively, likely because annotators were less willing (lazy) to point out each individual if there are too many of them. Thus the referring preference of the correct and distracting choices are similar in the RULE-PLURAL (both options prefer "they" to the person tags).

6.3.3 Shortcut effects on Adversarially-Modified Setting

We constructed the following validation sets to check the shortcut effects. ADVTOP-1 removes the most probable evidence (see Fig. 21), while in contrast KEEPTOP-K only uses the top-K potential pieces of evidence. Tab. 24 shows the results. Compared to STD VAL, ADVTOP-1 is more challenging since one important piece of evidence is masked out, thus performance drops by 14-32% accuracy on Q→A and 18-34% on QA→R. Given that the average length of the answer choices in both tasks are 7.65 and 16.19 tokens respectively, it is not understandable that masking out one token shall have such a big impact unless the models are fragile and base their decisions on single tokens. Finally, the strong performance in the KEEPTOP-K setting further shows models made decisions based on little facts instead of comprehensive thinking. For example, based on carefully chosen[1] three tokens, R2C is able to improve accuracy from 63.8% (full answers) to 65.9% (3-word answers). Note that we used a *single B2T2 model* (different initialization) to generate the same adversarial evaluation data *for all models*. This is why the performance drop is larger on B2T2 in Tab. 24.

6.3.4 Contribution of Our Robust Training

We next verify the extent to which robust training enables us to recover some of the lost performance. We train B2T2 based on the authors' reference implementation, but skip the expensive pre-training stage (contributing only 0.4% in [5]). We refer to this method as BASELINE, and compare it to the strategies described in Approach: Robust Training. Tab. 25 shows the results. First, we found using the rule-based and adversarial strategies (AUG RULE, AUG ADVTOP-1) to augment the training data achieved better performance in the corresponding evaluation settings (as expected), but did not perform well in the original nor the other modified setting. On the Q→A task (first 13 rows), when the probability of replacing a random token is small (e.g., MASKING 0.05), it leads to robust results in both

[1] The adversarial model used the label information to look for the token positions (see Eq. 15).

Table 24: Shortcuts in VCR: adversarially-modified evaluation.

	Method	Std Val	Rm. a shortcut	Utilizing the potential shortcuts		
			AdvTop-1	KeepTop-1	KeepTop-3	KeepTop-5
Q→A	R2C	63.8	49.8	51.8	65.9	67.5
	HGL	69.4	54.5	51.8	68.4	71.5
	TAB-VCR	69.9	54.9	49.6	65.1	69.7
	B2T2	68.5	37.0	51.0	75.0	80.4
QA→R	R2C	67.2	47.0	31.3	44.5	55.3
	HGL	70.6	51.6	33.7	48.8	60.2
	TAB-VCR	72.2	53.9	32.6	44.5	55.7
	B2T2	68.5	34.7	28.1	37.6	54.5

the original and rule-modified settings (69.3% vs. 68.5%, 63.9% vs 63.3%, etc.) However, performance degrades (64.1% v.s. 68.5%, 56.6% vs 63.3%) once too few pieces of evidence are used in training (Masking 0.30). Masking 0.10 + MLM slightly outperforms the baseline in some settings, but is worse than Masking 0.05. In contrast, our best curriculum learning method, Ours-CL Init0.30 Decay5e-5, outperforms all masking/MLM methods and the B2T2 baseline. We observe the benefit of dynamic, curriculum masking, compared to static masking from prior work, in both the original and modified settings.

6.3.5 Attention Weights Show Broader Use of Evidence

Next, we show that robust training leads to models' broader attention to various evidence. We use BertViz [264] and examine attention strength. In Fig. 22, we observe that to determine the effect of "turned around", Ours-CL (right) pays attention to more tokens in the question, and determines "walk away" to be important as the result of "turned around". In contrast, the baseline without robust training (middle) based the prediction of "turned" on "would" *because this content-free word* is in the question (a shallow match), thus did not

Table 25: Our method enables the most robust training. All results show Q→A except for the bottom two which show QA→R. The best method per group on Q→A is bolded, and the best method per task is underlined.

Method	Std Val	Rule-Singular	Rule-Plural	AdvTop-1
Baseline (B2T2)	68.5	63.3	65.3	37.0
Aug Rule	**67.0**	**78.8**	**69.9**	31.6
Aug AdvTop-1	64.4	57.3	57.0	**81.4**
Masking 0.05	**69.3**	**63.9**	**66.0**	48.8
Masking 0.10	68.7	62.8	64.7	50.1
Masking 0.15	68.2	62.0	63.3	**50.6**
Masking 0.30	64.1	56.6	56.8	47.5
Masking 0.05 + MLM	68.5	62.9	64.8	47.3
Masking 0.10 + MLM	69.1	63.8	65.0	**50.6**
Ours-CL Init0.30 Decay1e-4	69.6	64.5	64.7	51.7
Ours-CL Init0.30 Decay5e-5	**69.9**	**65.9**	**66.8**	54.5
Ours-CL Init0.50 Decay1e-4	69.4	65.0	65.0	53.0
Ours-CL Init0.50 Decay5e-5	69.8	65.4	66.3	**54.9**
Baseline (B2T2)	68.5	64.9	67.9	34.7
Ours-CL Init0.30 Decay5e-5	70.6	66.6	70.4	47.9

learn to *reason*.

Quantitatively, we compute the attention distribution on the validation set and the average *entropy* per BERT layer (from different attention heads and image examples). We show in Tab. 26 the ratio of entropy for Ours-CL vs Baseline. In the last layers (11 and 10), which are used to compute the answers, the entropy of Ours-CL is larger, which means our model pays attention to broader evidence.

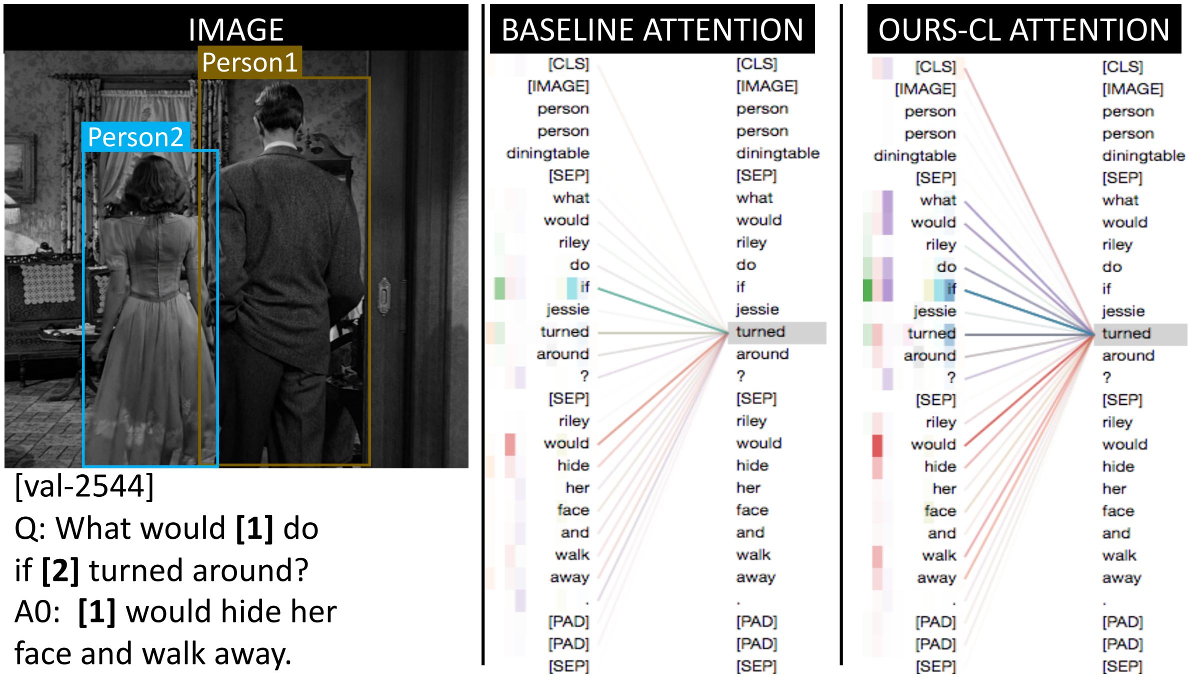


IMAGE
Person1
Person2
[val-2544]
Q: What would [1] do if [2] turned around?
A0: [1] would hide her face and walk away.
BASELINE ATTENTION
OURS-CL ATTENTION
[CLS] [IMAGE] person person diningtable [SEP] what would riley do if jessie turned around ? [SEP] riley would hide her face and walk away . [PAD] [PAD] [SEP]
[CLS] [IMAGE] person person diningtable [SEP] what would riley do if jessie turned around ? [SEP] riley would hide her face and walk away . [PAD] [PAD] [SEP]
[CLS] [IMAGE] person person diningtable [SEP] what would riley do if jessie turned around ? [SEP] riley would hide her face and walk away . [PAD] [PAD] [SEP]
[CLS] [IMAGE] person person diningtable [SEP] what would riley do if jessie turned around ? [SEP] riley would hide her face and walk away . [PAD] [PAD] [SEP]

Figure 22: Learned attention of the baseline and Ours-CL. Attention strength is denoted by darker/lighter shaded boxes under word "Layer". We show weights in BERT-Base Layer-9, which is potentially the last layer of interpretable high-level reasoning. In Layer-10, word features are aggregated in [SEP], while in the last Layer-11, [SEP] is gathered in [CLS]. Please zoom figure to 300%.

Table 26: Our model pays attention to broader evidence: The numbers shown are the ratios of attention entropies for Ours-CL and those corresponding to Baseline.

Entropy of	Layer11	Layer10	Layer9	Layer8	Layer7	Layer6
Ours-CL/Baseline	**104.62%**	**105.20%**	98.97%	98.07%	102.13%	101.83%
Entropy of	Layer5	Layer4	Layer3	Layer2	Layer1	Layer0
Ours-CL/Baseline	102.53%	99.61%	100.67%	101.52%	97.74%	98.93%

6.4 Conclusion

This chapter proved the thesis hypotheses H2 in a general setting, in which only vision and text modalities exist. We evaluated the effect of the observed shortcuts, i.e., shallow matching between questions and answers in the VCR dataset. This shows that some evidence can be unreliable — We demonstrated subtle changes to the answer options, which should not change the meaning or correct choice, do successfully trick methods, causing significant drops in performance for four recent models. We further proposed a novel technique for robust training, which applies masking on a curriculum, starting with a large amount of masking and gradually reducing it. We showed that our method was more successful in undoing the harmful effect of shortcuts, compared to techniques that have been previously used for achieving robustness through pre-training.

In the next chapter, we shall explore the unreliable multimodal supervision and still use the text captions as our research target. We will show that through a serials of processes of denoising, filtering, distillation, text captions can be used to learn reliable and robust object detection models.

Table 27: Conclusion - validated hypotheses in this chapter.

	Multimodal features help to understand images/videos with implicit persuasive intent, such as visual advertisements.	Text features can be unreliable if not modeled appropriately.	Text supervision contains noise, but can be used to localize visual objects in space, if modeled properly.	Text supervision provides contexts regarding visual objects, they are reliable cues for disambiguating entities and relations.	Noisy audio narrations as a multimodal signal can be modeled to localize video actions in temporal domain.
This Chapter		✓			

7.0 Cap2Det: Learning to Amplify Weak Caption Supervision for Object Detection

7.1 Introduction

Thus far, we discussed multimodal knowledge integration for visual reasoning and ways to use and model them efficiently, i.e., the utilization of multimodal inputs. Since the multimodal perception lies in both the inputs and supervision signals, we start to explore the next topic of using multimodal supervision from this chapter. The particular weakly supervised detection task we study is similar to human beings learning visual concepts from conversations. Much like visual reasoning, the major challenge is to learn robust and reliable models from noisy and sometimes unrelated data.

We are most interested in detecting visual objects, so we start from this chapter to learn object detectors from noisy captions. We will show in this chapter a series of processes to filter, amplify, and distill object knowledge from text captions. Then, Chapter 8 shall further utilize the object contexts from captions, to provide more reliable information regarding objects. Finally, Chapter 9 shall learn action detection model from multimodal cues in videos.

Learning to localize and classify visual is a fundamental problem in computer vision. It has a wide range of applications, including robotics, autonomous vehicles, intelligent video surveillance, and augmented reality. Since the renaissance of deep neural networks, object detection has been revolutionized by a series of groundbreaking works, including Faster-RCNN [211], Mask-RCNN [81] and YOLO [209]. Modern detectors can run in real-time on mobile devices, and have become the driving force for future technologies.

Despite these achievements, most modern detectors suffer from an important limitation: they are trained with expensive supervision in the form of large quantities of bounding boxes meticulously drawn by a large pool of human annotators. According to [19], the average annotation time per image is ***239.7 seconds***. Due to the well-known domain shift problem [38, 253, 137, 246, 76] and imperfect domain adaptation techniques, this means when detection is to be performed in a novel domain, the expensive annotation procedure needs to be repeated. Weakly supervised object detection (WSOD) techniques aim to alleviate the burden of collecting such expensive box annotations. The classic WSOD problem formulation [22, 275, 248, 247] treats an image as a bag of proposals, and learns to assign instance-level semantics to these proposals using multiple instance learning (MIL). WSOD has shown great potential for object detection, and the state-of-the-art model has reached 40% mAP [247] on Pascal VOC 2012. However, one critical assumption in WSOD is that the image-level label should be *precise*, indicating at least one proposal instance in the image needs to associate with the label. This assumption does not always hold especially for real-world problems and real-world supervision.

Weakly supervised detection methods need large-scale image-level object category labels. These labels require human effort that is provided in an unnatural, crowdsourced environment. However, more natural supervision for objects exists—for example, in the form of natural language descriptions that web users provide when uploading their photos to social media sites such as YouTube or Instagram. There are tens of millions of photos uploaded to Instagram every day, and a majority of them have titles, tags, or descriptions. Abundant videos with subtitles are similarly available on YouTube. These annotations are "free" in that no user was *paid* to provide them; they arise out of innate needs of users to make their content available to others.

However, existing WSOD methods cannot use such supervision. First, these natural language descriptions are unstructured; they need to be parsed and words relevant for object recognition need to be extracted, while non-object words are removed. Second, these descriptions are both imprecise and non-exhaustive—they might mention content that is not in the image (e.g. what event the user was attending or who they met after the photo was taken), and also omit content that actually is in the image but is not interesting. Consider the image in the bottom-right of Fig. 23. It contains numerous objects, many of which fairly large—e.g. dining table and bowls—yet the human providing the description did not mention these objects. Thus, directly feeding web data to the state-of-the-art WSOD system contains numerous limitations—at the very least, it under-utilizes the rich supervision that captions can provide.

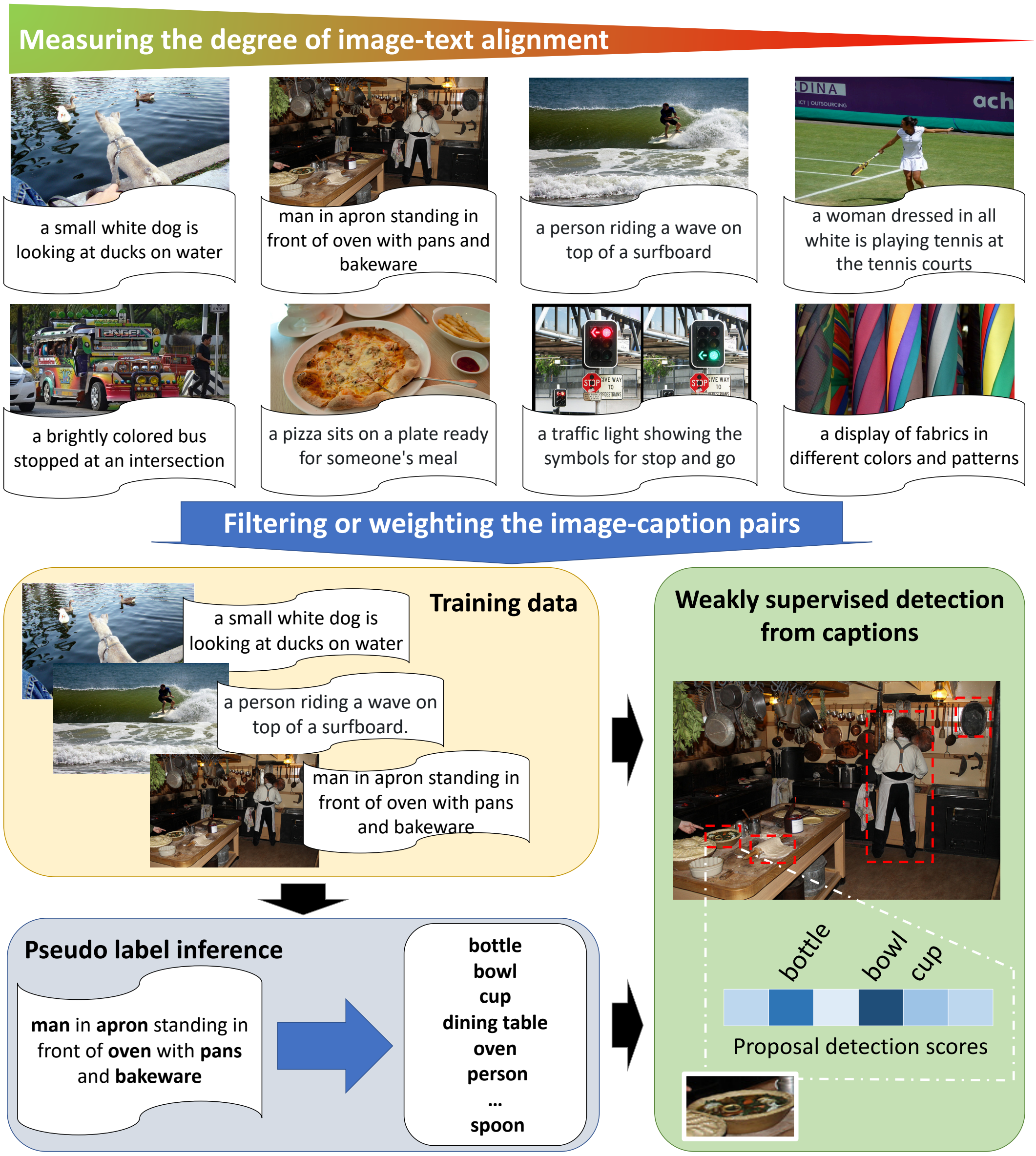

Measuring the degree of image-text alignment
a small white dog is looking at ducks on water
man in apron standing in front of oven with pans and bakeware
a person riding a wave on top of a surfboard
a woman dressed in all white is playing tennis at the tennis courts
a brightly colored bus stopped at an intersection
a pizza sits on a plate ready for someone's meal
a traffic light showing the symbols for stop and go
a display of fabrics in different colors and patterns
Filtering or weighting the image-caption pairs
Training data
a small white dog is looking at ducks on water
a person riding a wave on top of a surfboard.
man in apron standing in front of oven with pans and bakeware
Weakly supervised detection from captions
Pseudo label inference
man in apron standing in front of oven with pans and bakeware
bottle
bowl
cup
dining table
oven
person
...
spoon
bottle
bowl
cup
Proposal detection scores

Figure 23: An overview of our approach. We first determine the potential for strong object supervision signal from image-caption pairs (sorted from strongest to weakest signal). After selecting or weighting these pairs, we extract discrete object labels from the captions, and train a weakly-supervised object detection model with these pseudo labels at the image level. The model training part (bottom) is accepted by the ICCV [299], while we wrap the whole pipeline (+ filtering/weighting) in a journal under submission.

This chapter tackles the above challenge. We first explore the ways to filter captions to potentially benefit object detection models (Fig. 23 top). We adopt the scoring method proposed in [254] to estimate to what extent an image caption and the image provide overlapping information and propose two techniques to apply the scoring: one filters out noisy data, and the other one uses the scores to weigh object detection losses.

We then learns weakly supervised object detectors from images paired with noisy textual captions (Fig. 23 bottom). The key idea of applying such a multimodal cue as supervision is to bridge human-written free-form texts and visual objects. Our method relies on two key components. First, we train a textual classifier to map captions to discrete object labels. This classifier is not dataset-dependent, requires only a small set of labels, and generalizes beyond dataset boundaries. It enables us to bridge the gap between what humans mention in a caption, and what truly is in an image. Second, we use the pseudo ground truth labels predicted by this textual classifier, to train a weakly supervised object detection method. The method we propose extracts region proposals off-the-shelf, then for each proposal and each class, learns both a class score and a detection score. These scores are then refined using an iterative approach, to produce final detection results.

To summarize, our contributions are as follows:

- We propose a new task of learning visual concepts from noisy textual captions, a type of multimodal cues. Rather than treating object categories as IDs only, we also leverage their semantics, as well as synonyms of those object names.
- We show that we outperform alternative uses of captions, e.g. exactly matching the captions to object category words, or retrieving hand-annotated or predicted synonyms to the object categories from the captions. We show competitive WSOD performance by learning on COCO or Flickr30K captions. We further validate the benefit of our COCO-trained text classifier by applying it on Flickr30K, and leveraging training on Flickr30K then evaluating on PASCAL.
- In a side-by-side comparison under the classic WSOD setting, our model demonstrates superior performance with image-level supervision and achieves state-of-the-art performance on all three WSOD benchmarks (48.5% mAP@0.5 for VOC 2007, 45.1% mAP@0.5 on VOC 2012 and 23.4% mAP@0.5 on COCO).

- We demonstrate the success of explicitly modeling which image-caption pairs provide a strong signal for supervision, using new metrics that capture how closely the text follows the image. These metrics allow us to improve performance by up to 10%.

7.2 Approach

We train object detectors from supervision only consisting of noisy captions and corresponding images. In realistic scenarios, captions and images may contain complementary information. We hypothesize that even for crowdsourced, descriptive captions which closely follow the image (e.g. COCO), not all caption-image pairs provide equally strong supervision, as some captions will overlap with the image to a stronger degree. Figure 23 (top) shows example images sorted in descending order of image-text alignment, with the dog image and its caption being most aligned as all objects shown are also mentioned, while the bus, pizza, traffic and fabrics captions also contain concepts that are visually not shown or are visually ambiguous (intersection, meal, stop and go, display); hence extracting concrete nouns (objects) from these captions is more challenging. Thus, the first step in our framework is to automatically determine which image-caption pairs to use as supervision; we propose two alternative approaches, one which uses a hard cutoff over the image-text alignment score, and another which uses all image-caption pairs but gives them different weight. This part of the method is described in Section 7.2.1.

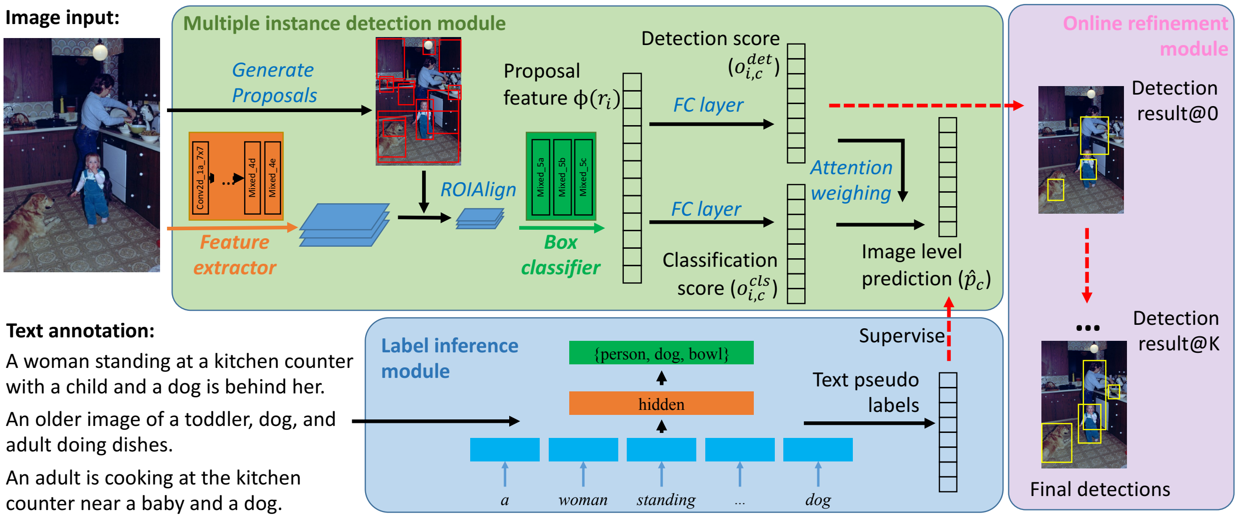
After selecting image-caption pairs for training, we next extract discrete labels at the image level (Section 7.2.2). We do so through a variety of techniques, the simplest of which is looking for exact string match between nouns in the caption and object words, and the most complex being training a classifier which takes in a caption (without a paired image) and maps this caption to a discrete set of labels (which may or may not be mentioned in the caption). Finally, given these pseudo ground-truth image-level labels, we train a variant of a prior weakly-supervised object detection technique: it first computes initial scores for each region and each object class, then refines these iteratively (Section 7.2.3). The overall architecture of training using image-level text annotations is shown in Fig. 24.

7.2.1 Filtering by Supervision Purity

We propose techniques for filtering image-caption pairs that are unlikely to be useful for training. The key idea is to estimate to what extent an image caption and the image provide overlapping (redundant) or complementary information. While complementarity is useful in general, for detection, we require redundancy, i.e., the same objects being both shown in the image and mentioned in the caption. We base our redundancy measurement on our group's prior work [254], which grants a score for each image-caption pair. This score measures how well-aligned the image and text modalities are, by capturing polysemy or multiple illustrations for the same semantic concept. In particular, if the images that co-occur with semantically similar texts are *not* visually similar, then either the texts have multiple meanings, or there are different ways to illustrate the same semantic concept. This diversity in illustration leads to low alignment between images and text.

Filtering. We hypothesize that selecting training data and filtering out noisy image-caption pairs will improve the detection model training. We provide an experiment in Sec. 7.3.4, which selects the 30,000 image-caption pairs from COCO that have the highest image-caption matching scores. As compared to random selection, our method provides significantly better detection results (Tab. 32).

Weighting. Hard-cutoff filtering requires finding the right cutoff value (e.g. top-30k), and it means discarding some potentially useful data. Compared to the filtering strategy, weighting does not require a hard cutoff and is more data-efficient. It applies different weights to image-caption pairs. For image-caption pairs that are more overlapped, weighting assigns large weight to the loss term in that these examples will likely be useful for training detection models. For image-caption pairs that are more complementary, weighting assigns small weights because the information may not well-aligned. In Eq. 21, we use normalized image-caption matching scores as the heuristic weighting factor to weigh different training examples. We provide an ablation in Sec. 7.3.4, and Tab. 33 shows the impact of using weighting mechanism.



A woman standing at a kitchen counter with a child and a dog is behind her.

An older image of a toddler, dog, and adult doing dishes.

An adult is cooking at the kitchen counter near a baby and a dog.

Figure 24: Cap2Det: harvesting detection models from free-form text. We propose to use a label inference module (bottom-center) to amplify signals in the free-formed texts to supervise the learning of the multiple instance detection network (top). The learned detection model is then refined by an online refinement module (right) to produce the final detection results.

7.2.2 Label Inference from Text

After getting the image-caption pairs estimated to be well-aligned, we now proceed to extract pseudo object labels from the selected noisy captions, to benefit weakly-supervised object detection. The foundation of WSOD builds on an important assumption from MIL (Eq. 19), which suggests that *precise* image-level labels should be provided. However, gathering such *clean* annotations is not trivial. In most real-life cases, the semantic counterpart for visual content appears in the form of natural language phrases, sentences, or even paragraphs (in newspapers), which is noisier than object labels.

The straightforward solution of extracting object labels from captions via lexical matching, does not work well. Consider an image with three sentence descriptions:

"*a* ***person*** *is riding a* ***bicycle*** *on the side of a bridge.*"

"*a* ***man*** *is crossing the street with his* ***bike***."

"*a* ***bicyclist*** *peddling down a busy city street.*"

However, only the first sentence exactly matches the categories “person” and “bicycle”. Even if we allow synonyms of “man” and “person” or “bicycle” and “bike”, only the first two precisely describe both objects, while the last one still misses the instance of “bicycle” unintentionally.

When using these examples to train object detectors, the first two instances may bring positive effect, but the last one will be wastefully discarded as false negative i.e. not relevant to the categories “person” or “bicycle”. Even worse, in the example shown in Fig. 23, none of the captions (one shown) mention the “bowls” or “spoons” that are present, and only some mention the “oven”.

This observation inspires us to amplify the supervision signal that captions provide, and squeeze more information out of them. Fig. 24 (bottom) shows the approach we use to amplify the signal. This text-only model takes free-form texts as input, embeds individual words to a 300D space using GloVe [200], and projects the embedded features to a 400D latent space. We then use max-pooling to aggregate the word-level representations. Then, we use this intermediate representation to predict the implied instances (e.g. 80 classes as defined in COCO, or any other categories); this prediction answers “what’s in the image” and serves as pseudo image-level labels in training object detectors.

It is worth noting that there exists a subtle balance when using pseudo labels to train object detectors. Admittedly, our strategy increases the recall rates thus more data could be utilized. However, with the increased recall, precision will drop inevitably thus the fundamental assumption in MIL is threatened. Specifically, the *precise label* assumption makes the model very sensitive to false positive cases: when inappropriate labels are given where none of the proposals have a good response, the model gets confused, resulting in non-optimal detections.

We finally adapt a two-steps procedure: first we look for an exact match of object labels from captions, following the intuition that *explicitly mentioned objects* should be significant and obvious enough in the image; second, when no object can be matched, we use our label inference model to predict labels as *unspoken intended objects* to guide the object detection. We show our method outperforms several strong alternatives that also infer pseudo labels.

Discussion. Our text classifier relies on both captions and category labels. However, once the bridge between captions and labels is established, this classifier generalizes to other datasets, as we show in Tab. 28. Importantly, we only need a small fraction of labels to train this text classifier; as we show in Fig. 25, precision ranges between 89% and 92% when we use between only 5% and 100% of the COCO data, while recall is stable at 62%. Thus, our text model could learn from a *single source* dataset with *a few* labels, then it could transfer the knowledge to other *target* datasets, requiring only free-form text as supervision.

7.2.3 Detection from Inferred Labels

We next describe how we use the inferred pseudo labels to train an object detection model. As shown in Fig. 24, we first extract proposals with accompanying features. An image is fed into the pretrained (on ImageNet [47]) convolutional layers. Then, *ROIAlign* [81] is used for cropping the proposals (at most 500 boxes per image) generated by *Selective Search* [259], resulting in fixed-sized convolutional feature maps. Finally, a box feature extractor is applied to extract a fixed-length feature for each proposal. If $[r_1, \dots, r_m]$ are the proposals of a given image x, this process results in proposal feature vectors $[\phi(r_1), \dots, \phi(r_m)]$ where each $\phi(r_i) \in \mathbb{R}^d$. Note that while our model is pretrained on ImageNet, it *does not leverage* any image labels at all on the datasets on which we train and evaluate our detection models (PASCAL and COCO).

Weakly Supervised Detection. We next introduce the prediction of image-level labels $\hat{p}_c$ ($c \in \{1, \dots, C\}$, where C is the number of classes) and of detection scores as a by-product. The proposal features $\phi(r_i)$ are fed into two parallel fully-connected layers to compute the detection scores $o^{\text{det}}_{i,c} \in \mathbb{R}^1$ (top branch in the green MIL module in Fig. 24) and classification scores $o^{\text{cls}}_{i,c} \in \mathbb{R}^1$ (bottom branch), in which both scores are related to a specific class c and the particular proposal r_i:

$$o^{\text{cls}}_{i,c} = w^{\text{cls}\intercal}_c \phi(r_i) + b^{\text{cls}}_c, \qquad o^{\text{det}}_{i,c} = w^{\text{det}\intercal}_c \phi(r_i) + b^{\text{det}}_c \tag{16}$$

We convert these scores into: (1) $p^{\text{cls}}_{i,c}$, the probability that object c presents in proposal

r_i; and (2) $p_{i,c}^{\text{det}}$, the probability that r_i is important for predicting image-level label y_c:

$$p_{i,c}^{\text{cls}} = \sigma(o_{i,c}^{\text{cls}}), \qquad p_{i,c}^{\text{det}} = \frac{\exp(o_{i,c}^{\text{det}})}{\sum_{j=1}^{m} \exp(o_{j,c}^{\text{det}})} \tag{17}$$

Finally, the aggregated image-level prediction is computed as follows, where greater values of $\hat{p}_c \in [0, 1]$ mean higher likelihood that c is present in the image:

$$\hat{p}_c = \sigma\left(\sum_{i=1}^{m} p_{i,c}^{\text{det}} o_{i,c}^{\text{cls}}\right) \tag{18}$$

Assuming the label $y_c = 1$ if and only if class c is present, the **m**ultiple **i**nstance **d**etection loss used for training the model is defined as:

$$L_{\text{mid}} = -\sum_{c=1}^{C} \left[y_c \log \hat{p}_c + (1 - y_c) \log(1 - \hat{p}_c) \right] \tag{19}$$

Preliminary detection scores. The weakly supervised detection score given both proposal r_i and class c is the product of $p_{i,c}^{\text{cls}}$ and $p_{i,c}^{\text{det}}$ which is further refined as described in *Online Instance Classifier Refinement.*

Online Instance Classifier Refinement. The third component of our WSOD model is Online Instance Classifier Refinement (OICR), as proposed by Tang *et al.* [248]. The main idea behind OICR is simple: Given a ground-truth class label, the top-scoring proposal, as well as proposals highly overlapping with it, are selected as references. These proposals are treated as positive examples for training the box classifier of this class while others are treated as negatives. The initial top-scoring proposal may only partially cover the object, so allowing highly-overlapped proposals to be treated as positives gives them a second chance to be considered as containing an object, in the subsequent model refinement. This reduces the chance of propagating incorrect predictions. In addition, sharing the convolutional features between the original and refining models makes the training more robust.

Following [248], we stack multiple refining classifiers and use the output of the previous one to generate instance-level supervision to train the successor. The detection score at the 0-th iteration is computed using $s_{i,c}^{(0)} = p_{i,c}^{cls} p_{i,c}^{det}, s_{i,C+1}^{(0)} = 0$ (where $C + 1$ is the background class). Given the detection score $s_{i,c}^{(k)}$ at the k-th iteration, we use the image-level label to get the *instance-level* supervision $y_{i,c}^{(k+1)}$ at the $(k+1)$-th iteration. Assume that c' is a label attached to image x, we first look for the top-scoring box r_j ($j = \arg\max_i s_{i,c'}^{(k)}$). We then let $y_{i,c'}^{(k+1)} = 1, \forall i \in \{l | IoU(r_l, r_j) > \textit{threshold}\}$. When $k > 0$, $s_{i,c}^{(k)}$ is inferred using a $(C+1)$-way FC layer, as in Eq. 16. The OICR training loss is defined in Eq. 20.

$$L_{\text{oicr}}^{k} = -\frac{1}{m}\sum_{i=1}^{m}\sum_{c=1}^{C+1} \hat{y}_{i,c}^{(k)} \log s_{i,c}^{(k)}, \qquad k = 1, \ldots, K \tag{20}$$

Unlike the original OICR, our WSOD module aggregates logits instead of probability scores, which in our experience stabilizes training. We also removed the reweighing of untrustworthy signals emphasized in [248] since we found it did not contribute significantly.

The final loss we optimize is Eq. 21. We refine our model for 3 times ($K = 3$) if not mentioned otherwise.

$$L = L_{\text{mid}} + \sum_{k=1}^{K} L_{\text{oicr}}^{k} \tag{21}$$

7.3 Experiments

We evaluate all components of our method: the text classifier that learns to map captions to object labels, the weakly supervised detection module, and the refinement. We show that compared to alternative strategies, our approach extracts the most accurate and expansive information from the captions (Sec. 7.3.2). By training on COCO captions, we achieve close to state-of-the-art results on weakly supervised detection on PASCAL, even though the supervision we leverage is weaker than competitor methods. Importantly, our text classifier allows us to excel at the task of training on Flickr30K to detect on PASCAL, even though that classifier was trained on a different dataset (COCO). We show our approach outperforms prior methods on the task of learning from image-level labels (Sec. 7.3.3). Finally, we show the improvements achieved by filtering and weighing noisy image-caption examples (Sec. 7.3.4). We conclude that the redundancy between image and text is key to train a successful weakly supervised detection model.

7.3.1 Implementation Details

Before training the detector, we use [254] to measure the redundancy between the image and text and offline compute four scores for each image-text pair. We use Selective Search [259] from OpenCV [23] to extract at most 500 proposals for each image. We follow the "Selective search quality" parameter settings in [259]. We prefer Selective Search because it is a generic, dataset-independent proposal generation procedure, as opposed to other CNN-based alternatives which are trained end-to-end from a specific dataset in a supervised fashion. We also experimented with Edge Boxes [329] but got inferior performance. We use TensorFlow [1] as our training framework. To compute the proposal feature vectors, we use the layers ("Conv2d_1a_7x7" to "Mixed_4e") from Inception-V2 [244] to get the conv feature map, and the layers ("Mixed_5a" to "Mixed_5c") from the same model to extract the proposal feature vectors after the ROIAlign [81] operation. The Inception-V2 model is pretrained on ImageNet [47]; the supervised detector counterpart of our model, using this architecture, was explored by [89]. To augment the training data, we resize the image randomly to one of the four scales $\{400, 600, 800, 1200\}$. We also randomly flip the image left to right at training time. At test time, we average the proposal scores from the different resolution inputs. We set the number of refinements to 3 for the OICR since it gives the best performance. For post-processing, we use non-maximum-suppression with IoU threshold of 0.4. We use the AdaGrad optimizer, a learning rate of 0.01, and a batch size of 2 as commonly used in WSOD methods [248, 247]. The models are usually trained for 100K iterations on Pascal VOC (roughly 40 epochs on VOC2007 and 17 epochs on VOC2012) and 500K on COCO (8.5 epochs), using a validation set to pick the best model. Our implementation is available at `https://github.com/yekeren/Cap2Det`.

7.3.2 Using Captions as Supervision

In this section, we first evaluate our method (Sec. 7.2.2 - 7.2.3), including our proposal for how to squeeze the most information out of the weak supervision that captions provide (Sec. 7.2.2). We also experiment with alternative strategies of generating pseudo labels, and evaluate the performance in terms of precision and recall by comparing with ground-truth labels. We shall leave the evaluation of our filtering component in the last experiement (Sec. 7.3.4).

Alternative Strategies. We compared with multiple pseudo-label generation baselines when lexical matching (ExactMatch) fails to find a match. As previous examples show, considering synonyms can effectively reduce off-target matching rates. Thus our first baseline adopts a *manually constructed, hence expensive* COCO synonym vocabulary list (ExtendVocab) which maps 413 words to 80 categories [155]. Another variant, GloVePseudo, takes advantage of GloVe word embeddings [200], assigning pseudo-labels for a sentence by looking for the category that has the smallest embedding distance to any word in the sentence. We also follow a similar strategy with [294] to finetune the GloVe word embeddings on COCO using a visual-text ranking loss, and use the pseudo labels retrieved by the resultant LearnedGloVe as a stronger baseline. The final reference model of using ground-truth image-level labels GT-Label is an upper bound. Note that apart from the strategy used to mine image-level labels, these strategies all use the same architecture and WSOD approach as our method (Sec. 7.2.3). In later sections, we show combinations of the exact match strategy with these methods (when exact match fails), resulting in EM+GloVePseudo, EM+LearnedGloVe, EM+ExtendVocab and EM+TextClsf. We examine how well these and other strategies leverage captions from COCO and Flickr30K [305] to produce accurate detection.

Analysis of Textual Supervision. In Fig. 25 we show the *precision* and *recall* of these label inference methods evaluated directly on the COCO image-level labels (5,000 examples of the *val2017* set). We observe that ExtendVocab, which uses the handcrafted word-synonyms dictionary, provides the best recall (60.6%) among all methods but provides the worst precision of 81.1%. The word-embedding-based top-scoring matching methods of GloVePseudo and LearnedGloVe provide precise predictions (84.5% and 84.7% respectively, which are the highest). However, our TextClsf achieves significantly improved precision compared to these. We would like to point out that while in Tab. 28 and 29, our method uses the full COCO training set (118,287 concatenated captions), it achieves very similar performance with even a small fraction of the data. With 5% of the data, the method achieves 89% precision (vs 92% precision with 100% of the data), both of which are much higher than any other baselines; recall is about 62% for both 5% and 100% training data. In other words, it is sufficient to use a small portion of precise text labels to train a generalizable label inference classifier, and the knowledge can transfer to other datasets as we show in Tab. 28.

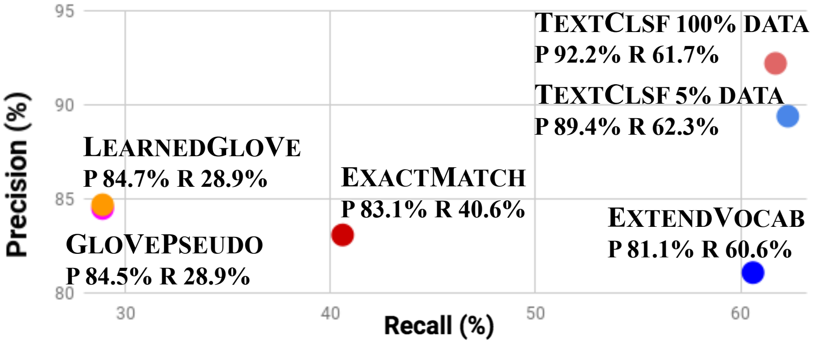


Figure 25: Analysis of different text supervision. We compare the pseudo labels (Sec. 7.2.2) to COCO *val* ground-truth.

Table 28: Average precision (in %) on the VOC 2007 test set (learning from COCO and Flickr30K captions). We learn the detection model from the COCO captions describing the 80 objects, but evaluate on only the overlapping 20 VOC objects.

Methods	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv	mean
Training on different datasets using ground-truth labels:																					
GT-LABEL VOC	68.7	49.7	53.3	27.6	14.1	64.3	58.1	76.0	23.6	59.8	50.7	57.4	48.1	63.0	15.5	18.4	49.7	55.0	48.4	67.8	48.5
GT-LABEL COCO	65.3	50.3	53.2	25.3	16.2	68.0	54.8	65.5	20.7	62.5	51.6	45.6	48.6	62.3	7.2	24.6	49.6	34.6	51.1	69.3	46.3
Training on COCO dataset using captions:																					
EXACTMATCH (EM)	63.0	**50.3**	50.7	25.9	**14.1**	64.5	50.8	33.4	17.2	49.0	48.2	46.7	44.2	59.2	10.4	14.3	49.8	37.7	21.5	47.6	39.9
EM + GLOVEPSEUDO	**66.6**	43.7	53.3	29.4	13.6	65.3	**51.6**	33.7	15.6	50.7	46.6	45.4	47.6	**62.1**	8.0	**15.7**	48.6	46.3	30.6	36.4	40.5
EM + LEARNEDGLOVE	64.1	49.9	**58.6**	24.9	13.2	**66.9**	49.2	26.9	13.1	57.7	**52.8**	42.6	**53.2**	58.6	14.3	15.0	45.2	50.3	34.1	43.5	41.7
EM + EXTENDVOCAB	65.0	44.9	49.2	**30.6**	13.6	64.1	50.8	28.0	**17.8**	59.8	45.5	**56.1**	49.4	59.1	16.8	15.2	**51.1**	**57.8**	14.0	**61.8**	42.5
EM + TEXTCLSF	63.8	42.6	50.4	29.9	12.1	61.2	46.1	**41.6**	16.6	**61.2**	48.3	55.1	51.5	59.7	**16.9**	15.2	50.5	53.2	**38.2**	48.2	**43.1**
Training on Flickr30K dataset using captions:																					
EXACTMATCH (EM)	**46.6**	**42.9**	42.0	9.6	7.7	31.6	44.8	53.2	13.1	28.0	39.1	43.2	31.9	**52.5**	4.0	**5.1**	38.0	28.7	**15.8**	41.1	31.0
EM + EXTENDVOCAB	37.8	37.6	35.5	11.0	**10.3**	18.0	47.9	51.3	**17.7**	25.5	37.0	47.9	**35.2**	46.1	**15.2**	0.8	27.8	35.6	5.8	42.0	29.3
EM + TEXTCLSF	24.1	38.8	**44.5**	**13.3**	6.2	**38.9**	**49.9**	**60.4**	12.4	**47.4**	**39.2**	**59.3**	34.8	48.1	10.7	0.3	**42.4**	**39.4**	14.1	**47.3**	**33.6**

To better understand the generated labels, we show two qualitative examples in Fig. 26. The image on the right shows that our model infers "tie" from the observation of "presenter", "conference" and "suit", while all other methods fail to extract this object category for visual

Table 29: COCO test-dev results (learning from COCO captions). We report these numbers by submitting to the COCO evaluation server. The best method is shown in **bold**.

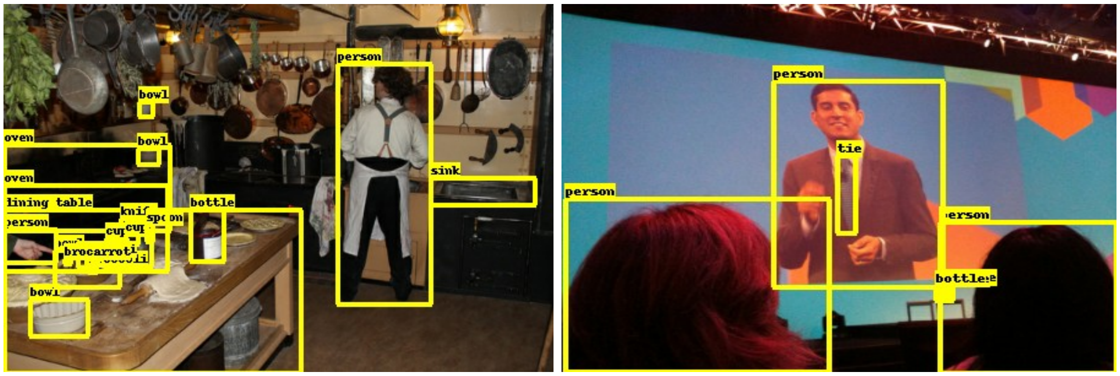
Methods	Avg. Precision, IoU			Avg. Precision, Area		
	0.5:0.95	0.5	0.75	S	M	L
GT-Label	10.6	23.4	8.7	3.2	12.1	18.1
ExactMatch (EM)	8.9	19.7	7.1	2.3	10.1	16.3
EM + GloVePseudo	8.6	19.0	6.9	2.2	10.0	16.0
EM + LearnedGloVe	8.9	19.7	7.2	2.5	10.4	**16.6**
EM + ExtendVocab	8.8	19.4	7.1	2.3	10.5	16.1
EM + TextClsf	**9.1**	**20.2**	**7.3**	**2.6**	**10.8**	**16.6**

detection. We argue the capability of inferring reasonable labels from captions is critical for learning detection model from noisy captions.

Training with COCO Captions. We next train our detection model using the COCO captions [36]. We use the 591,435 annotated captions paired to the 118,287 *train2017* images. For evaluation, we use the COCO *test-dev*2017 and PASCAL VOC 2007 *test* sets. In our supplementary file, we show qualitative examples from the COCO *val* set.

Tab. 28 shows the results on PASCAL VOC 2007. At the top are two upper-bound methods that train on image-level *labels*, while the rest of the methods train on image-level *captions*. ExactMatch (EM) performs the worst probably due to its low data utilization rate, as evidenced by the fact that all methods incorporating pseudo labels improve performance notably. Specifically, EM+GlovePseudo uses free knowledge of the pre-trained GloVe embeddings. It alleviates the synonyms problem to a certain extent, thus it improves the mAP by 2% compared to ExactMatch. However, the GloVe embedding is not optimized for the specific visual-captions, resulting in noisy knowledge transformation. EM+LearnedGloVe learns dataset-specific word embeddings. Its performance, as expected, is 3% better than EM+GloVePseudo in terms of mAP. The strongest baseline is

EM+ExtendVocab, as the manually picked vocabulary list covers most frequent occurrences. However, collecting such vocabulary requires human effort, and is not a scalable and transferable strategy. Our EM+TextClsf outperforms this expensive baseline, especially for categories "cat", "cow", "horse", and "train".



A man is in a kitchen making pizzas .
Man in apron standing on front of oven with pans and bakeware .
A baker is working in the kitchen rolling dough .
A person standing by a stove in a kitchen .
A table with pies being made and a person standing near a wall with pots and pans hanging on the wall .

GroundTruth: dining table, oven, person, bottle, bowl, broccoli, carrot, cup, knife, sink, spoon
ExactMatch: **dining table**, **oven**, **person**
ExtendVocab: **dining table**, **oven**, **person**, pizza
GloVePseudo: **oven**
LearnedGloVe: **dining table**
TextClsf: **person**, **oven**, **bowl**, **dining table**, **bottle**, **cup**, **spoon**, **knife**, chair, refrigerator, pizza

A presenter projected on a large screen at a conference
People watching an on screen presentation of a gentleman in a suit .
People watch a man delivering a lecture on a screen .
A large screen showing a person wearing a suit
An audience is looking at an film of a man taking that is projected onto a wall .

GroundTruth: person, tie, bottle
ExactMatch: **person**
ExtendVocab: **person**
GloVePseudo: **person**
LearnedGloVe: **person**
TextClsf: **person**, **tie**, chair, handbag, tv

Figure 26: Demonstration of different pseudo labels. Our method fills the gap between what is present and what is mentioned, by making inferences on the semantic level. Matches to the ground truth are shown in blue.

At the top of Tab. 28 are two upper-bound methods which rely on ground-truth image-level captions. Despite the noisy supervision, our EM+TextClsf almost bridges the gap to the COCO-labels upper bound.

For the results on COCO (Tab. 29), the gaps in performance between the different methods are smaller, but as before, our proposed EM+TextClsf shows the best performance. We believe the smaller gaps are because many of the COCO objects are not described precisely via natural language, and the dataset itself is more challenging than PASCAL thus gain may be diluted by tough examples.

Qualitative results on COCO. We provide qualitative examples on the COCO *val* set. We compare the ExactMatch and our EM+TextClsf side-by-side in Fig. 27. Qualitatively, our proposed method EM+TextClsf provides better detection results than the baseline ExactMatch. Thus, we conclude that it has squeezed more useful and precise information than the ExactMatch baseline.

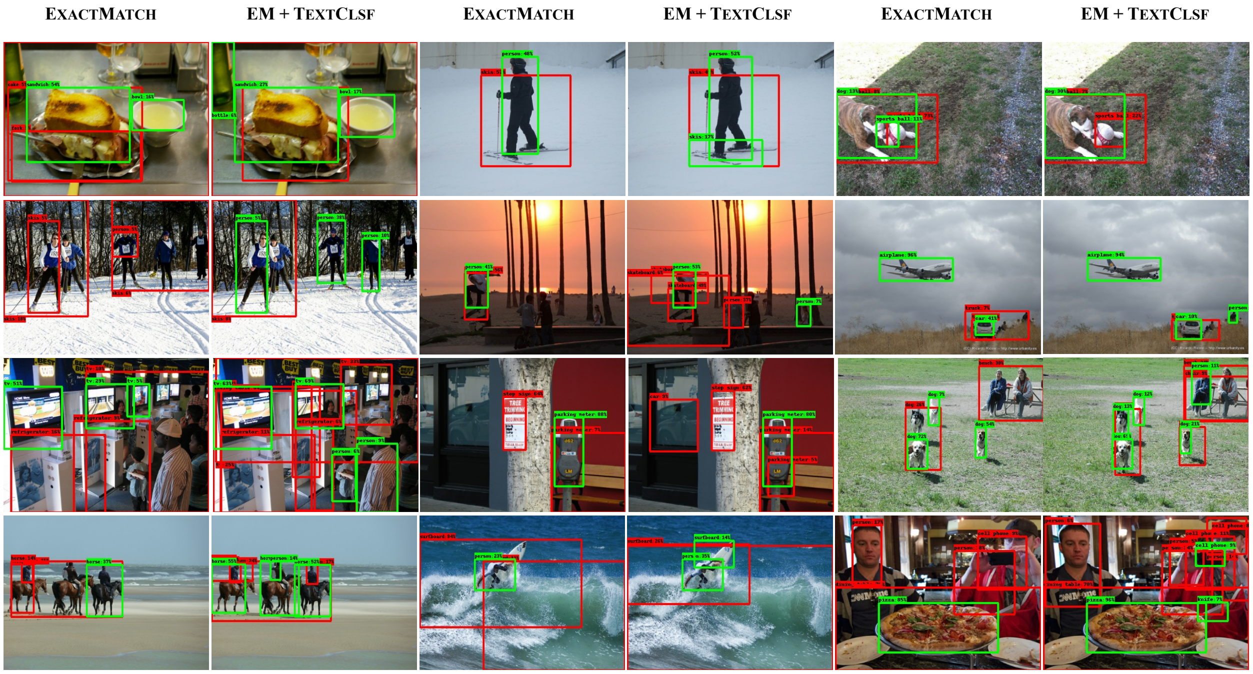


Figure 27: Visualization of our Cap2Det model results on COCO *val* set. We show boxes with confidence scores $> 5\%$. Green boxes denote correct detection results ($IoU > 0.5$) while red boxes indicate incorrect ones. Best viewed with 300% zoom-in.

Training with Flickr30K Captions. We also train our model on the Flickr30K dataset [305], which contains 31,783 images and 158,915 descriptive captions. Training on Flickr30K is more challenging: on one hand, it includes less data compared to COCO; on the other hand, we observe that the recall rate of the captions is only 48.9% with ExactMatch which means only half of the data can be matched to some class names. The results are shown in the bottom of Tab. 28. We observe that due to the limited training size, the detection models trained on Flickr30K captions achieve weaker performance than those trained on COCO captions. However, given the "free" supervision, the 33.6% mAP is still very encouraging. Importantly, we observe that even though our text classifier is trained on COCO captions and labels, it generalizes well to Flickr30K captions, as evidenced by the gap between EM+TextClsf and EM+ExtendVocab.

Data v.s. Performance. We show the potential of our model using Flickr30K and MIRFlickr1M [92]. For the latter, we concatenate the title and all user-generated content tags to form caption annotation. We then use our text classifier learned on COCO to rule out examples unlikely to mention our target classes. This filtering results in a dataset with around 20% of the original data, and we refer to it as Flickr200K. We use 10%, 20%, 50%, 100% data from both datasets, and report average precision on VOC 2007. We see from Fig. 28 that as training data increases, mAP increases accordingly. To estimate model potential, we fit a square root function to the rightmost four points in the figure and use it to estimate 54.4 mAP at 1 million samples.

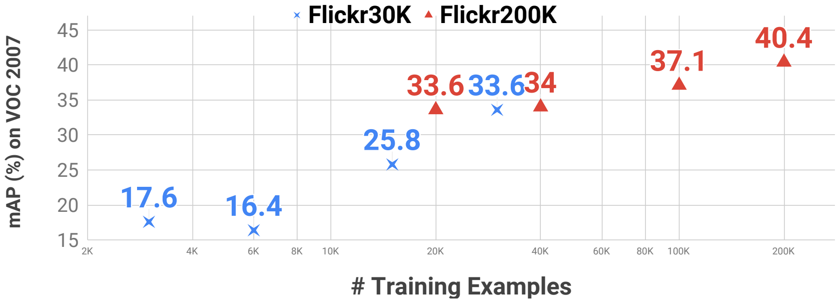


Figure 28: Data v.s. Performance. Our text classifier learned on COCO generalized well on Flickr30K and the noisier Flickr200K data formed by user-generated content tags.

7.3.3 Using Image Labels as Supervision

We finally show the performance of our method in the classic WSOD setting where *image-level supervision* is available. These results validate the method component described in Sec. 7.2.3. They also serve as an approximate *upper bound* for the more challenging task in Sec. 7.3.2.

Results on PASCAL VOC. For each image, we extract object categories from all the ground-truth bounding boxes, and only keep these *image-level* labels for training, discarding box information. For VOC 2007 and 2012, we train on 5,011 and 11,540 *trainval* images respectively and evaluate on 4,952 and 10,991 *test* images.[1] We report the standard mean Average Precision (mAP) at IoU > 0.5. We compare against multiple strong WSOD baselines. The results are shown in Tab. 30, and our single model outperforms the baseline methods (sometimes even ensemble methods) by a large margin. On VOC 2007, our model improves the mAP of the state-of-the-art single method TS^{2}C method by 9%. On VOC 2012, our method outperforms the strongest single-model baseline PCL-OB-G VGG16 by 11%. Some prior work uses their WSOD detection results to further train an Fast-RCNN [69] detector (denoted as "+FRCNN" in Tab. 30) and obtain an additional 3 to 4 percents improvements on mAP. Even without such post-processing or ensemble, our model still achieves competitive performance on both VOC 2007 and 2012.

Table 30: Average precision (in %) on the Pascal VOC test set using image-level labels. The top shows VOC 2007 and the bottom shows VOC 2012 results. The best single model is in **bold**, and best ensemble in *italics*.

Methods	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv	mean
VOC 2007 results:																					
OICR VGG16 [248]	58.0	62.4	31.1	19.4	13.0	**65.1**	62.2	28.4	24.8	44.7	30.6	25.3	37.8	65.5	15.7	**24.1**	41.7	46.9	**64.3**	62.6	41.2
PCL-OB-G VGG16 [247]	54.4	**69.0**	39.3	19.2	**15.7**	62.9	**64.4**	30.0	**25.1**	52.5	44.4	19.6	39.3	**67.7**	**17.8**	22.9	46.6	**57.5**	58.6	63.0	43.5
TS^{2}C [275]	59.3	57.5	43.7	27.3	13.5	63.9	61.7	59.9	24.1	46.9	36.7	45.6	39.9	62.6	10.3	23.6	41.7	52.4	58.7	56.6	44.3
OICR Ens.+FRCNN [248]	65.5	67.2	47.2	21.6	22.1	68.0	68.5	35.9	5.7	*63.1*	49.5	30.3	*64.7*	66.1	13.0	*25.6*	*50.0*	57.1	60.2	59.0	47.0
PCL-OB-G Ens.+FRCNN [247]	63.2	*69.9*	47.9	22.6	*27.3*	*71.0*	*69.1*	49.6	12.0	60.1	*51.5*	37.3	63.3	63.9	15.8	23.6	48.8	55.3	61.2	62.1	*48.8*
Ours	**68.7**	49.7	**53.3**	**27.6**	14.1	64.3	58.1	**76.0**	23.6	**59.8**	**50.7**	**57.4**	**48.1**	63.0	15.5	18.4	**49.7**	55.0	48.4	**67.8**	**48.5**
VOC 2012 results:																					
OICR VGG16 [248]	67.7	61.2	41.5	25.6	22.2	54.6	49.7	25.4	19.9	47.0	18.1	26.0	38.9	67.7	2.0	22.6	41.1	34.3	37.9	55.3	37.9
PCL-OB-G VGG16 [247]	58.2	**66.0**	41.8	24.8	**27.2**	55.7	**55.2**	28.5	16.6	**51.0**	17.5	28.6	**49.7**	**70.5**	7.1	**25.7**	47.5	36.6	44.1	**59.2**	40.6
TS^{2}C [275]	67.4	57.0	37.7	23.7	15.2	**56.9**	49.1	64.8	15.1	39.4	19.3	**48.4**	44.5	67.2	2.1	23.3	35.1	40.2	**46.6**	45.8	40.0
OICR Ens.+FRCNN [248]	71.4	69.4	55.1	29.8	*28.1*	55.0	*57.9*	24.4	17.2	*59.1*	21.8	26.6	57.8	71.3	1.0	23.1	*52.7*	37.5	33.5	56.6	42.5
PCL-OB-G Ens.+FRCNN [247]	69.0	*71.3*	*56.1*	30.3	27.3	55.2	57.6	30.1	8.6	56.6	18.4	43.9	*64.6*	*71.8*	7.5	23.0	46.0	44.1	42.6	58.8	44.2
Ours	**74.2**	49.8	**56.0**	**32.5**	22.0	55.1	49.8	**73.4**	**20.4**	47.8	**32.0**	39.7	48.0	62.6	**8.6**	23.7	**52.1**	**52.5**	42.9	59.1	**45.1**

Effects of the Basic Network and OICR. The performance gain in our model comes from the following two aspects: (1) a more advanced detection model backbone architecture and (2) the online instance classifier refinement (OICR). Fig. 29 shows the performance of

[1] VOC 2012 result: `http://host.robots.ox.ac.uk:8080/anonymous/NOR9IV.html`

our method and that of Tang *et al.* [248] (OICR VGG_M), both refining for 0, 1, 2, 3 times. With no (0) refinement, our basic network architecture outperforms the VGG_M backbone of Tang *et al.* by 27% in mAP. But the basic architecture improvement is not sufficient to achieve top results. If we use OICR to refine the models 1, 2, or 3 times, we gain 24%, 29%, and 30% respectively while Tang *et al.*achieve smaller improvement (22%, 28%, and 29% gains).

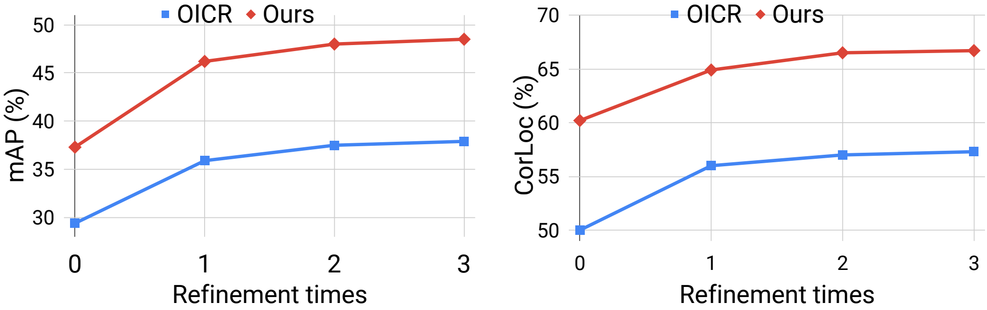


Figure 29: Analysis of our basic network and OICR components on VOC 2007. Comparison of the performance of our model and OICR VGG_M after iterative refinement.

Results on COCO. We train our model on the 118,287 *train2017* images, using the image-level ground truth labels. We report mAP at IoU=.50:.05:.95 and mAP@0.5, on the 20,288 *test-dev2017* images. We compare to a representative fully-supervised detection model [211]; "Faster Inception-V2" [89] which is our method's supervised detection counterpart, and a recent WSOD model [247]. As demonstrated in Tab. 31, our model outperforms the previous state-of-the-art WSOD method (PCL-OB-G Ens + FRCNN) by 15% in terms of mAP, but the gap between general WSOD methods (including ours) and the supervised methods is still large due to the disparate supervision strength.

7.3.4 Impact of Filtering Noisy Captions

We next show that the potential purity of objects mentioned is the key to train a good weakly supervised object detector. We validate the two proposed methods in Sec. 7.2.1:

Detection results using image-caption filtering. We use a limited 30,000 image-caption pairs (a subset) from COCO *train2017* split for training, assuming a setting of

Table 31: COCO detection using image-level labels, with supervised detection models at the top, best WSOD in **bold**.

Methods	Avg. Precision, IoU	
	0.5:0.95	0.5
Faster RCNN [211]	21.9	42.7
Faster Inception-V2 [89]	28.0	-
PCL-OB-G VGG16 [247]	8.5	19.4
PCL-OB-G Ens.+FRCNN [247]	9.2	19.6
Ours	**10.6**	**23.4**

restricted computation resources and training time. To trim the large amount of data, we keep the most useful examples while removing the others. Specifically, we use the metrics of *homogeneity* and *symmetry* to measure the image-caption relevance, which inherited from [254]'s *diversity* and *discrepancy* but take the reverse because we want to assign high scores to image-caption pairs that mention the same objects. Homogeneity measures how similar the images paired with a text are visually. Symmetry measures cycle consistency: how close the neighbor-of-neighbors of an image or text sample are to the original query.

The higher the *homogeneity* and *symmetry*, the better alignment between the image and text, and more likely the captions describe the visual objects in detail. We use random sampling of 30K examples as a baseline. Hom-Image, Hom-Text, Sym-Image, Sym-Text use both the *homogeneity* and *symmetry* metrics applying on both the image and text modalities to filter examples, accordingly.

Tab. 32 shows the results. We see that the performances of ExtendVocab, GloVe, and TextClsf are improved (in most cases) using the filtered training data. If we use a random selection of 30K examples, the performances are 36.7%, 38.6%, 40.4%, respectively. Using *image homogeneity score* (Hom-Image) for filtering improved these methods by 9% (40.1% v.s. 36.7%), 7% (41.3% v.s. 38.6%), 0.2% (40.5% v.s. 40.4%) while using Sym-

Table 32: Comparing the filtering strategies with the random sampling baseline, using AP (in %) on VOC 2007 test.

Label inference / Im-cap scoring	ExtendVocab	GloVe	TextClsf
Random	36.7	38.6	40.4
Hom-Image	40.1	41.3	40.5
Hom-Text	40.4	40.8	39.9
Sym-Image	40.6	40.2	41.2
Sym-Text	38.6	37.9	37.9

Image improved 11% (40.6% v.s. 36.7%), 4% (40.2% v.s. 38.6%), 2% (41.2% v.s. 40.4%). Besides, we find that the filtering helps more for the ExtendVocab and GloVe while seems to be not that helpful for TextClsf. We suspect the reason is that TextClsf had already explained the gap between the image and text thus is not sensitive to the improved filtered training data. However, this text classifier requires some small number of ground-truth labels. In contrast, Hom-Image, Hom-Text and Sym-Image with Glove achieve competitive results to the basic TextClsf (with Random), but do not require any labels. Thus, homogeneity and symmetry could be used to determine which captions provide strong supervision for object detection, without the need for *any* ground-truth labels.

Results using image-caption weighting. One weakness of the filtering approach is that it requires a hard cutoff of the dataset examples. In comparison, weighting applies a soft "cutoff" to the data. It never drops data, thus is data-efficient. We use the Home-Image score as the per-example weighting factor in Eq. 21.

Tab. 33 shows the results. The top shows the performance of filtering approaches. Since the filtering strategy had to trade-off between the image-text relevance and efficient data utilization, it is not easy to find the perfect balance. It shows that even with a good filtering strategy (e.g., Hom-Image Filtering (30K)), using 30K "clean" training examples is still

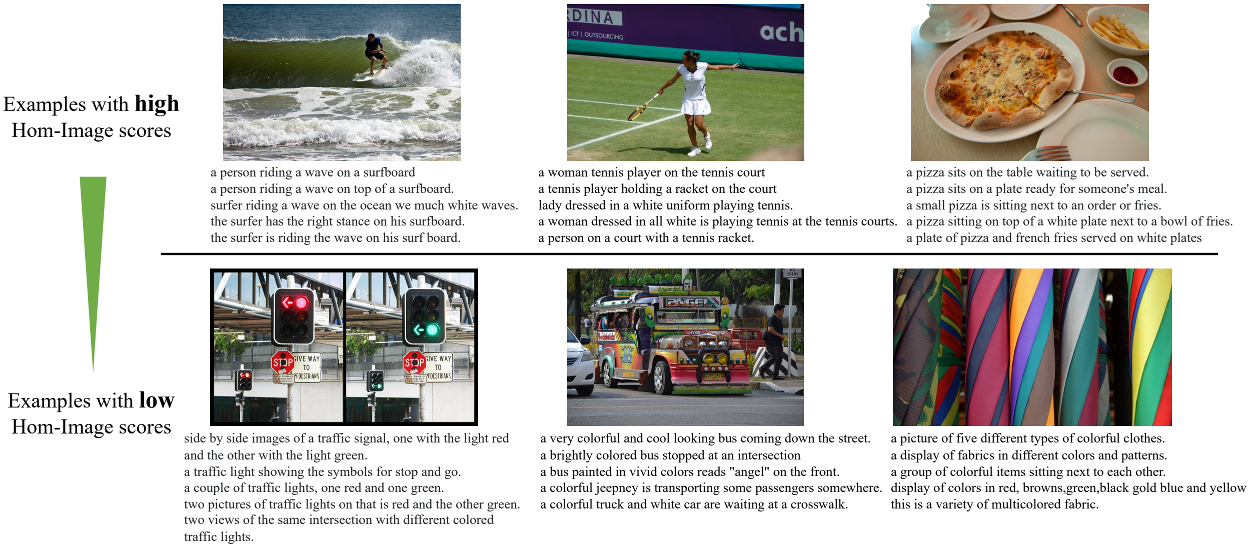
Table 33: Comparing the weighting strategy with the filtering alternates, using AP (in %) on VOC 2007 test.

Im-cap scoring \ Label inference	ExtendVocab	GloVe	TextClsf
Random (30K)	36.7	38.6	40.4
Hom-Image Filtering (30K)	40.1	41.3	40.5
No weighting (118K)	42.5	40.5	**43.1**
Hom-Image Weighting (118K)	**43.5**	**42.6**	42.2

inferior than training on the full COCO dataset (No weighting (118K)), for two of the three label inference methods (columns). However, if we apply the Hom-Image *weighting* on the loss term, the performance (Hom-Image Weighting (118K)) is generally improved (43.5% v.s. 42.5%, 42.6% v.s. 40.5%, 42.2% v.s. 43.1%), except for TextClsf which requires annotations.

This shows that homogeneity computation, which requires no ground-truth labels, can be used to boost the performance of the text classifier, without discarding any of the original data. This is an important finding with important ramifications for multimodal learning. Approaches to learn visual representations have benefited greatly from widely available videos with narrations, and our method suggests how the useful signal and the noise in such data can be distinguished to boost the quality of the learned representations, without requiring annotations.

Image-caption pairs with high/low scores. In Fig. 30, we observe that image-caption pairs with high homogeneity scores usually have a simple background and feature a single object in the center. In contrast, the images with low homogeneity scores are usually more complicated, not all objects shown are mentioned, and mentioned concepts may be abstract. Thus, we qualitatively proved that the homogeneity scores measure the relevance and redundancy between the image and text modalities. It helps to rule out less useful


Examples with **high** Hom-Image scores
Examples with **low** Hom-Image scores

a person riding a wave on a surfboard
a person riding a wave on top of a surfboard.
surfer riding a wave on the ocean we much white waves.
the surfer has the right stance on his surfboard.
the surfer is riding the wave on his surf board.

a woman tennis player on the tennis court
a tennis player holding a racket on the court
lady dressed in a white uniform playing tennis.
a woman dressed in all white is playing tennis at the tennis courts.
a person on a court with a tennis racket.

a pizza sits on the table waiting to be served.
a pizza sits on a plate ready for someone's meal.
a small pizza is sitting next to an order or fries.
a pizza sitting on top of a white plate next to a bowl of fries.
a plate of pizza and french fries served on white plates

side by side images of a traffic signal, one with the light red and the other with the light green.
a traffic light showing the symbols for stop and go.
a couple of traffic lights, one red and one green.
two pictures of traffic lights on that is red and the other green.
two views of the same intersection with different colored traffic lights.

a very colorful and cool looking bus coming down the street.
a brightly colored bus stopped at an intersection
a bus painted in vivid colors reads "angel" on the front.
a colorful jeepney is transporting some passengers somewhere.
a colorful truck and white car are waiting at a crosswalk.

a picture of five different types of colorful clothes.
a display of fabrics in different colors and patterns.
a group of colorful items sitting next to each other.
display of colors in red, browns,green,black gold blue and yellow
this is a variety of multicolored fabric.

Figure 30: Image-caption pairs with high homogeneity scores on the top, and low scores on the bottom.

examples to better train a detector.

7.4 Conclusion

In this chapter, we proved the thesis hypotheses H3 (see Tab. 34). We showed how we could successfully leverage naturally arising, weak supervision in the form of captions. We explicitly deal with noise in the captions and propose the filtering solution based on the supervision purity metrics (Sec. 7.2.1). Furthermore, we amplify the signal that captions provide by learning to bridge the gap between what human annotators mention, and what is present in the image (Sec. 7.2.2). Both solutions provide ways for training a robust weakly supervised object detetion model. As compared to using the ideal image-labels requiring human labor, we show the difference in terms of performance is small (40.4 v.s. 48.5), and our method can benefit from more data hence further be improved.

One weakness regarding the current approach is that we merely turn captions into class vectors while ignoring the abundant information hidden in the textual structures. In the next chapter, we explore textual structures (planned to use constituency parsing and context-free grammars) to use them to: (1) better disambiguate the referring objects (e.g., "girl in a hat" v.s. "man in white shirt"); (2) analyze the relationship between objects (e.g., "a man holding a baseball bat"). We expect the textual structures to make the use of text supervision more reliable.

Table 34: Conclusion - validated hypotheses in this chapter.

	Multimodal features help to understand images/videos with implicit persuasive intent, such as visual advertisements.	Text features can be unreliable if not modeled appropriately.	Text supervision contains noise, but can be used to localize visual objects in space, if modeled properly.	Text supervision provides contexts regarding visual objects, they are reliable cues for disambiguating entities and relations.	Noisy audio narrations as a multimodal signal can be modeled to localize video actions in temporal domain.
This Chapter			✓		

8.0 Linguistic Structures as Weak Supervision for Visual Scene Graph Generation

8.1 Introduction

The previous chapter provides a simple way to learn visual object detectors using the captions. However, the proposed Cap2Det model has two disadvantages: (1) it did not fully utilize captions in that it pays attention to only the amplified entities; (2) the entities focused on may be unreliable, if not considering the contexts (Fig. 31 shows people under different contexts). In this chapter, we additionally detect the relations among entities in a weakly supervised manner and attempt to strengthen the robustness of our model using the enlightening textual contexts. We aim at the joint task of object detection (OD) and visual relation detection (VRD) in the weakly supervised scenario — weakly supervised scene graph generation (WS-SGGen). We found that captions provide enough hints to solve both the OD and VRD tasks.

We start our discussion with an observed discrepancy. While scene graphs are a holistic, contextual representation of an image, the types of supervision that have been used capture context in an impoverished way. In particular, prior methods use supervision in the form of either subject-predicate-object triplets with bounding boxes for the subject and object [151, 187, 289] or subject-predicate-object triplets at the image level only [313, 317]. Thus, information in the supervision is local (separate triplets) while the scene graph to be output captures the entire image. This discrepancy between the properties of the desired output (global) and training data (local) becomes problematic due to potential ambiguity in the visual input. For example, in Fig. 31, multiple *persons* are standing on the *rails.* Thus, standard supervision (top) which breaks down a scene graph into triplets, may create confusion.

In contrast, *captions* capture global context that allows us to link multiple triplets, and localize a man who is both standing on the rails, and wearing a (checkered) shirt. Captions are linguistic constructs, and language could be argued to capture common sense (e.g.,

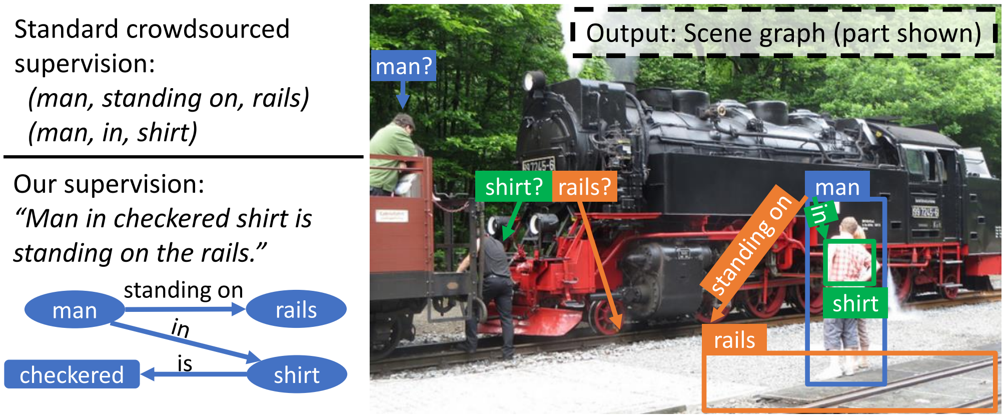

Standard crowdsourced supervision:
(man, standing on, rails)
(man, in, shirt)
Our supervision:
"Man in checkered shirt is standing on the rails."
man
standing on
rails
in
checkered
is
shirt
Output: Scene graph (part shown)
man?
shirt?
rails?
man
in
standing on
shirt
rails

Figure 31: We tackle the problem of generating scene graphs with supervision in the form of captions at training time. Parsing from captions enables utilization of the huge amount of image-text data available on the internet. The linguistic structure extracted maintains the relational information described in the caption without the loss of cross-triplet references, and facilitates disambiguation.

BERT [50] models are good at question-answering and commonsense tasks). Captions are also advantageous in terms of cost: humans naturally provide language descriptions of visual content they upload, thus caption-like supervision can be seen as "free". However, caption supervision contains noise, which presents some challenges. First, captions provide supervision at the image level, similar to prior work in weakly-supervised scene graph generation [313]. Second, prior work [175, 299] shows that captions do not cover all relevant objects: not all content is mentioned, and some of the mentioned content is not referring to the image explicitly or is not easily localizable. Because captions are noisy, the supervision we use is even weaker than prior work [313].

In this chapter, we propose an approach that leverages global context, using captions as supervision. Our approach models context for scene graphs in two ways. First, it extracts information from captions beyond the subject-predicate-object entities (e.g., in the form of attributes like "checkered", in Fig. 31). This context enables more accurate representations of concepts, and thus more accurate localization of each subject-predicate-object triplet. Second, visuo-linguistic context provides a way to reason about common-sense relationships within each triplet, to prevent non-sensical triplets from being generated (e.g., "rails standing on man" is unlikely, while "man standing on rails" is likely). To cope with the challenges of the noise contained in captions, we rely on an iterative detection method which helps prune some spurious relations between caption words and image regions, via boostrapping. While the captions we use are crowdsourced, our method paves the road for using image-caption pairs harvested from the internet for free, using text accompanying images on the web, from blogs, social media posts, YouTube video descriptions, and instructional videos [172, 226, 305]. Note that our method internally uses a graph with broad types of nodes, including adjectives, even though these are not part of the graph that is being output at test time. A side contribution is an adaptation of techniques from weakly-supervised object detection to improve localization of subject and object through iterative refinement, which has not been used for scene graph generation before.

To isolate the contribution of global context from the noise contained in captions (i.e., objects not being mentioned), we verify our approach in two settings. First, we construct a ground-truth triplet graph by connecting triplets with certain overlap. We show that our full method greatly outperforms prior work (it boosts the performance of [313] by 59%-67%). Second, we use two types of actual captions. This causes overall performance to drop, but we observe that modeling phrasal (cross-triplet) and sequential (within-triplet) linguistic context achieves strong results, significantly better than more direct uses of captions, and competitive with methods using clean image-level supervision.

To summarize, our contributions are as follows:

- We examine a new mechanism for scene graph generation using a new type of weak supervision.
- We contextualize embeddings for subject/object entities based on linguistic structures (e.g. noun phrases).
- We propose new joint classification and localization of subject, object and predicate within a triplet.
- We leverage weakly-supervised object detection techniques to improve scene graph generation.

8.2 Approach

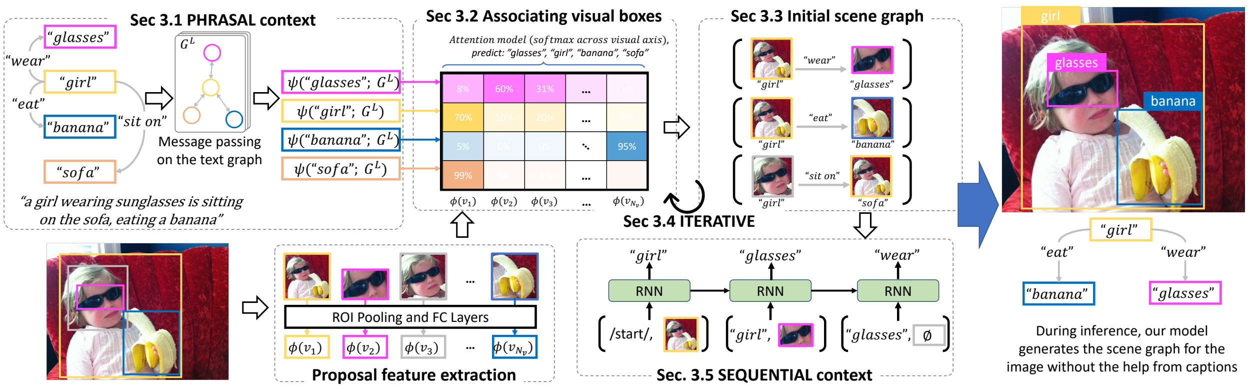

Sec 3.1 PHRASAL context
"glasses"
"wear"
"girl"
"eat"
"banana"
"sit on"
"sofa"
G^L
Message passing on the text graph
"a girl wearing sunglasses is sitting on the sofa, eating a banana"
ψ("glasses"; G^L)
ψ("girl"; G^L)
ψ("banana"; G^L)
ψ("sofa"; G^L)
Sec 3.2 Associating visual boxes
Attention model (softmax across visual axis), predict: "glasses", "girl", "banana", "sofa"
8%
60%
31%
...
1%
70%
10%
20%
0%
5%
0%
95%
99%
1%
$\phi(v_1)$
$\phi(v_2)$
$\phi(v_3)$
$\phi(v_{N_v})$
Sec 3.3 Initial scene graph
"girl"
"wear"
"glasses"
"eat"
"banana"
"sit on"
"sofa"
Sec 3.4 ITERATIVE
RNN
/start/,
"girl",
"glasses",
∅
Sec. 3.5 SEQUENTIAL context
ROI Pooling and FC Layers
Proposal feature extraction
girl
glasses
banana
During inference, our model generates the scene graph for the image without the help from captions

Figure 32: Model overview. Our model uses the image's paired caption as weak supervision to learn the entities in the image and the relations among them. At inference time, it generates scene graphs without help from texts. To learn our model, we first allow context information to propagate on the text graph to enrich the entity word embeddings (Sec. 8.2.1). We found this enrichment provides better localization of the visual objects. Then, we optimize a text-query-guided attention model (Sec. 8.2.2) to provide the image-level entity prediction and associate the text entities with visual regions best describing them. We use the joint probability (Eq. 27) to choose boxes associated with both subject and object (Sec. 8.2.3), then use the top scoring boxes (Eq. 28) to learn better grounding (Sec. 8.2.4). Finally, we use an RNN (Sec. 8.2.5) to capture the vision-language common-sense and refine our predictions. Our code is available at `https://github.com/yekeren/WSSGG`.

Inputs. Our method does not rely on dense human-annotated instances and relations, but takes in linguistic structures as supervised signals (Fig. 32 top-left). Such structural text information is abandoned in other weakly-supervised methods [313, 316, 317]. We first convert captions paired with images into text graphs using a language parser [219]. The resulting graphs describe the entities in the caption and the relations (e.g., verbs or prepositions) among them. We call this setting Cap-Graph. Our method's performance depends on how exhaustive the caption is, and how robust is the parser chosen. Thus, we also design a setting where we extract a ground-truth text graph from the scene graph annotations, ignoring bounding boxes (GT-Graph).

Table 35: Overview of notation for the visual features, linguistic structure G^L and supervision parsed from G^L.

Visual features		
V_{prop}	Region proposals	$n_v \times 1$
V_{feat}	Region proposal features	$n_v \times d_{cnn}$
$n_v = 20$	Number of region proposals	
$d_{cnn} = 1536$	Feature dimension	
Text graph $G^L(E, R)$, parsed from caption		
$E = [e_i]_{i=1}^{n_e}$	Entities (graph nodes)	$\|E\| = n_e$
$R = [(r_i, s_i, o_i)]_{i=1}^{n_r}$	Relations (graph edges)	$\|R\| = n_r$
n_e, n_r	Number of entities/relations in a graph	
c_e, c_r	Number of entity/relation classes (vocab size)	
e_i	The i-th entity node, $e_i \in \{1 \cdots c_e\}$	
r_i	The i-th relation edge, $r_i \in \{1 \cdots c_r\}$	
s_i, o_i	Subject/object index of i-th relation, $s_i, o_i \in \{1 \cdots n_e\}$, e_{s_i}, e_{o_i} refer to subject/object	
Frozen GloVe embeddings		
W_{ent}	Entity embedding matrix	$c_e \times d$
W_{rel}	Relation embedding matrix	$c_r \times d$
Image-level labels parsed from G^L		
Y_{ent}	$Y_{ent}[i, :]$ is the one-hot representation of e_i	$n_e \times c_e$
Y_{rel}	$Y_{rel}[i, :]$ is the one-hot representation of r_i	$n_r \times c_r$
Y_{cssub}, Y_{csobj}	$Y_{cssub}[i, :], Y_{csobj}[i, :]$ are one-hot repr of e_{s_i}, e_{o_i}	$n_r \times c_e$
Y_{cspred}	Alias of Y_{rel}	$n_r \times c_r$
Instance-level pseudo labels		
n_t	Number of iterations to improve $\boldsymbol{g}$	
$\boldsymbol{g}^{(t)}, t \in \{0 \cdots n_t\}$	Grounding vector, if E=[*girl*, *banana*], $\boldsymbol{g}$=[10, 17] means proposal v_{10} is class *girl* and v_{17} is *banana*	$n_e \times 1$
$Y_{det}^{(t)}, t \in \{0 \cdots n_t\}$	Entity detection label, $Y_{det}[i, j]$=1 means the proposal v_i involves the j-th entity class	$n_v \times c_e$
Y_{relsub}, Y_{relobj}	Relation detection label, $Y_{relsub}[i, j]$=1 means the proposal v_i may serve as a subject, and can apply the j-th relation to an unknown object; $Y_{relobj}[i, j]$=1 means the proposal v_i may serve as an object, some unknown subject can apply the j-th relation to v_i	$n_v \times c_r$

Training pipeline overview (Fig. 32): We extract the visual object proposals using FasterRCNN [211]. We extract the text graph from paired captions (Cap-Graph) or directly read the ground-truth text graph (GT-Graph). We use a graph neural network based on the phrasal structure to enrich the text node representation (Fig. 32 top-left, Sec. 8.2.1). This enrichment simplifies the later localization step because we can search for more specifically described regions (e.g., "girl eating banana," rather than "girl"). By optimizing the image-level entity scores and treating the text entities as queries, we obtain attention scores, which strongly imply the visual regions that best describe the text entities (Fig. 32 top-middle, Sec. 8.2.2). We design a way to learn from the weak signal of the attention scores and predict initial relation detection results in the form of 5-tuples (Sec. 8.2.3). These groundings are further refined using WSOD techniques [248] (Sec. 8.2.4). Finally, we capture visuo-linguistic common sense to further rule out unlikely relation tuples (Fig. 32 middle-bottom, Sec. 8.2.5). We use an RNN to model the fluency of scene graph tuples, enforcing that subject/object regions should be followed by their labels, and subject/object should be followed by object/predicate. This module reassigns labels and reranks 5-tuples to improve the relation detection: if an uncommon tuple is fed to the model, it will be assigned a low score.

8.2.1 Modeling Phrasal Context

We first determine how to represent the text entities to be matched in the image. A naive solution would be to use the word embeddings, but this method ignores the context captured in phrases. We advocate the use of the hints in the phrasal structure, namely mentions of related adjectives and objects. As shown in Fig. 32 top-left, “wearing sunglasses,” “sitting on the sofa” and “eating a banana” provide context for the same “girl” and make her distinguishable from other potential instances of “girl”. We infer the contextualized entity word features via the phrasal context and apply them in Sec. 8.2.2 to localize visual objects.

We have summarized all notations in Tab. 35 to facilitate reading the following text. The linguistic structure (Fig. 32 top-left) parsed from a caption is represented using a text graph $G^L = (E, R)$. $E = [e_1 \cdots e_{n_e}]^T$ denotes the n_e text graph entities where each $e_i \in \{1 \dots c_e\}$ represents an entity class ID (c_e classes in total, which are defined by [313] or [284] in our experiments; in Fig. 32 top-left, $E =$ [“glasses”, “girl”, “banana”, “sofa”]^T). $R = [(r_1, s_1, o_1) \cdots (r_{n_r}, s_{n_r}, o_{n_r})]^T$ describes the n_r relations. For the i-th relation: $r_i \in \{1 \dots c_r\}$ is the relation class ID; $s_i, o_i \in \{1 \dots n_e\}$ are entity indices: e_{s_i} denotes the subject entity and e_{o_i} the object entity; in Fig. 32 top-left, $R =$ {(“wear”, 2, 1), (“eat”, 2, 3), (“sit”, 2, 4))}. Given the GloVe embedding [200] of the entity and relation classes $W_{ent} \in \mathbb{R}^{c_e \times d}$, $W_{rel} \in$

$\mathbb{R}^{c_r \times d}$, and the one-hot representation of entities and relations $Y_{ent} \in \mathbb{R}^{n_e \times c_e}$, $Y_{rel} \in \mathbb{R}^{n_r \times c_r}$ (each row is a c_e or c_r-dim one-hot vector, and there are n_e and n_r rows, respectively), the initial entity and relation word embeddings can be represented as $H^{(0)}_{ent} = Y_{ent} W_{ent} \in \mathbb{R}^{n_e \times d}$ and $H^{(0)}_{rel} = Y_{rel} W_{rel} \in \mathbb{R}^{n_r \times d}$.

Now we compute phrasal contextualized entity embeddings $\psi(E; G^L) \in \mathbb{R}^{n_e \times d}$. Alg. 1 shows the process, and can be stacked several times. We update relation edge embeddings, then aggregate the relation features into the connected entity nodes, using linear layers ϕ^r and ϕ^α applied on the concatenation of inputs. We use $\psi(E; G^L) = H^{(t)}_{ent}, (t > 1)$ in the next section, to localize visual entities.

Algorithm 1: Message passing to utilize phrasal context. We use GraphNets [17] to implement.

```
Input  : Text graph $G^L = (E, R)$
         Initial entity features $H^{(t)}_{ent} = [\boldsymbol{e}_1, \ldots, \boldsymbol{e}_{n_e}]^T$
         Initial relation features $H^{(t)}_{rel} = [\boldsymbol{r}_1, \ldots, \boldsymbol{r}_{n_r}]^T$
Output: Updated $H^{(t+1)}_{ent}$, $H^{(t+1)}_{rel}$
for $i \leftarrow 1$ to $n_r$ do
    $\boldsymbol{r}'_i \leftarrow \phi^r(\boldsymbol{r}_i, \boldsymbol{e}_{s_i}, \boldsymbol{e}_{o_i})$  // Update edge, $\boldsymbol{r}'_i \in \mathbb{R}^{d \times 1}$
    $\alpha_i \leftarrow \phi^\alpha(\boldsymbol{r}_i, \boldsymbol{e}_{s_i}, \boldsymbol{e}_{o_i})$  // Update edge weight, $\alpha_i \in \mathbb{R}^1$
for $i \leftarrow 1$ to $n_e$ do
    $\boldsymbol{e}'_i \leftarrow \sum_{j=1:n_r, o_j=i} \{ \frac{\exp(\alpha_j)}{\sum_{k=1:n_r, o_k=i} \exp(\alpha_k)} \} \boldsymbol{r}'_j$  //Aggregate, $\boldsymbol{e}'_i \in \mathbb{R}^{d \times 1}$
return $H^{(t+1)}_{ent} = [\boldsymbol{e}'_1 \cdots \boldsymbol{e}'_{n_e}]^T, H^{(t+1)}_{rel} = [\boldsymbol{r}'_1 \cdots \boldsymbol{r}'_{n_r}]^T$
```

8.2.2 Associating Text Entities with Visual Boxes

After getting the contextualized entity embeddings $\psi(E; G^L) \in \mathbb{R}^{n_e \times d}$, we seek their associated visual regions $\boldsymbol{g}^{(0)} \in \mathbb{R}^{n_e \times 1}$ (i.e., grounding vector), where each $\boldsymbol{g}^{(0)}_i$ ranges in $\{1 \cdots n_v\}$ and $v_{\boldsymbol{g}^{(0)}_i}$ denotes the visual box best describing the text entity e_i. We obtain $\boldsymbol{g}$ using an attention mechanism. By optimizing the image-level prediction, we expect the model to learn to focus on the most informative and distinguishable regions, which can often be used as instance references for training object detectors.

We first project $V_{feat} \in \mathbb{R}^{n_v \times d_{cnn}}$ to the d-dim visual-language space, resulting in attention and classification heads $H_{att}, H_{cls} \in \mathbb{R}^{n_v \times d}$. Then, we compute $D_{dot} \in \mathbb{R}^{n_e \times n_v}$, in which $D_{dot}[i, j]$ measures the compatibility between text entity e_i and visual region v_j. We softmax-normalize D_{dot} to get the attention matrix $A^{(0)} \in \mathbb{R}^{n_e \times n_v}$, and obtain $\boldsymbol{g}^{(0)}$ by selecting the max-valued entry.

$$\begin{aligned}
&H_{att} = V_{feat} W_{att},\ H_{cls} = V_{feat} W_{cls} \\
&D_{dot} = \psi(E; G^L) H_{att}^T, \qquad A^{(0)}[i, j] = \frac{\exp(D_{dot}[i, j])}{\sum_{k=1}^{n_v} \exp(D_{dot}[i, k])} \\
&\boldsymbol{g}_i^{(0)} = \underset{j \in \{1 \cdots n_v\}}{\operatorname{argmax}} A^{(0)}[i, j] \qquad (22)
\end{aligned}$$

We use image-level entity labels $Y_{ent} \in \mathbb{R}^{n_e \times c_e}$ as supervision to learn proper attention scores. We first aggregate the image-level weighted visual features $F = [\boldsymbol{f}_1 \cdots \boldsymbol{f}_{n_e}]^T \in \mathbb{R}^{n_e \times d}$, where $\boldsymbol{f}_i$ denotes the image-level feature encoded with proper attention to highlight text entity e_i. For example, given $e_i =$ "glasses" in Fig. 32, the model needs to shift attention to the glasses visual region by adjusting the i-th row of $A^{(0)}$. The final image-level entity classification score is given by $P_{cls} \in \mathbb{R}^{n_e \times c_e}$, and the grounding module is trained using cross-entropy.

$$\begin{aligned}
&F = A^{(0)} H_{cls},\ F' = F W_{ent}^T \\
&P_{cls}[i, j] = \frac{\exp(F'[i, j])}{\sum_{k=1}^{c_e} \exp(F'[i, k])} \qquad (23)
\end{aligned}$$

$$L_{grd} = -\sum_{i=1}^{n_e} \sum_{j=1}^{c_e} Y_{ent}[i, j] \log P_{cls}[i, j] \qquad (24)$$

8.2.3 Initial Scene Graph Generation

Thus far, the text entity embeddings $H_{ent}^{(0)}$ played a role in the grounding procedure, and so did the one-hot encoded label Y_{ent} extracted from the caption. Next, the model learns to predict the entities and relations without help from captions, which will not be available at inference time.

To this end, given entities $E = [e_1 \cdots e_{n_e}]^T$, relations $R = [(r_1, s_1, o_1) \cdots (r_{n_r}, s_{n_r}, o_{n_r})]^T$, and grounded boxes $[v_{\boldsymbol{g}_1^{(0)}} \cdots v_{\boldsymbol{g}_{n_e}^{(0)}}]^T$, we first parse the *target* instance labels. We extract $Y_{det}^{(0)} \in \mathbb{R}^{n_v \times c_e}$ and $Y_{relsub}, Y_{relobj} \in \mathbb{R}^{n_v \times c_r}$ using Eq. 25, in which all non-mentioned matrix entries are set to 0. $Y_{det}^{(0)}[i,j] = 1$ means visual region v_i involves the j-th entity class. $Y_{relsub}[i,j] = 1$ denotes the potential subject visual region v_i (e.g. a "person" region) may apply the j-th relation (e.g. "ride") to an unknown object. $Y_{relobj}[i,j] = 1$ denotes an unknown subject may apply the j-th relation to the potential object visual region v_i (e.g. a "horse" region). We add $_{rel}$ to highlight Y_{relsub}, Y_{relobj} are relation instance-level labels, but are attached to the grounded subject and object visual boxes respectively.

$$\begin{aligned}
&Y_{det}^{(0)}[i,j] = 1 \text{ if } \exists k \in \{1 \cdots n_e\}, s.t.(\boldsymbol{g}_k^{(0)} = i, e_k = j) \\
&Y_{relsub}[i,j] = 1 \text{ if } \exists k \in \{1 \cdots n_r\}, s.t.(\boldsymbol{g}_{s_k}^{(n_t)} = i, r_k = j) \\
&Y_{relobj}[i,j] = 1 \text{ if } \exists k \in \{1 \cdots n_r\}, s.t.(\boldsymbol{g}_{o_k}^{(n_t)} = i, r_k = j)
\end{aligned} \tag{25}$$

We next learn to predict the instance-level labels based on these targets, using entity detection head $H_{det}^{(0)} \in \mathbb{R}^{n_v \times d}$, and relation detection heads $H_{relsub}, H_{relobj} \in \mathbb{R}^{n_v \times d}$. Then, we matrix-multiply the three heads to the entity embedding $W_{ent} \in \mathbb{R}^{c_e \times d}$ and relation embedding $W_{rel} \in \mathbb{R}^{c_r \times d}$, and softmax-normalize, resulting in entity detection scores $P_{det}^{(0)} \in \mathbb{R}^{n_v \times c_e}$ and subject/object detection scores $P_{relsub}, P_{relobj} \in \mathbb{R}^{n_v \times c_r}$. We use cross-entropy loss terms $L_{det}^{(0)}, L_{relsub}, L_{relobj}$ similar to Eq. 24 to approximate $P_{det}^{(0)} \sim Y_{det}^{(0)}$, $P_{relsub} \sim Y_{relsub}$, and $P_{relobj} \sim Y_{relobj}$.

$$\begin{aligned}
&X \in \{det, relsub, relobj\}, \qquad W' \in \{W_{ent}, W_{rel}\} \\
&H_X = V_{feat} W_X, \qquad F_X = H_X W'^T \\
&P_X[i,j] = \frac{\exp(F_X[i,j])}{\sum_k \exp(F_X[i,k])}
\end{aligned} \tag{26}$$

After training the aforementioned model, we can detect entities using $P_{det}^{(0)} \in \mathbb{R}^{n_v \times c_e}$ and detect relations using $P_{rel} \in \mathbb{R}^{n_v \times n_v \times c_r}$, where $P_{rel}[i,j,k] = \min(P_{relsub}[i,k], P_{relobj}[j,k])$. Intuitively, we treat the relation as valid if it could be both implied from the subject and object visual regions. For example, if the model infers “ride” from the “person” region and estimates “ride” can also apply to object region “horse”, it determines that “ride” is the proper predicate bridging the two regions. [313, 317] proposed similar architectures to infer relation from a single region, [313] for optimizing runtime and [317] to avoid bad solutions. We use this idea because it is simple and effective, in combination with our stronger module in Sec. 8.2.5.

Test time post-processing. Given $P_{det}^{(0)}$, and P_{rel}, we adopt the *top-K predictions* (in experiments, k=$50, 100$) denoted in Eq. 27 as the initial scene graph generation (SGGen) results. In Eq. 27, the universal set $U = \{(v_{s_i^v}, v_{o_i^v}, s_i^e, p_i^r, o_i^e)\}_i$ denotes all possible 5-tuple combinations and B is a subset of U of size k. The goal is to seek the subset $B (B \subset U$ and $|B| = k)$ such that the sum of log probabilities is maximized. Within a specific B, $s^v, o^v \in \{1 \cdots n_v\}$ are the indices of proposal boxes to represent the subject and object regions, respectively; $s^e, o^e \in \{1 \dots c_e\}$ are subject and object entity class IDs; $p^r \in \{1 \dots c_r\}$ is the relation class ID. To implement Eq. 27 in practice, we use non-max suppression on $P_{det}^{(0)}$ to reduce the search space (ruling out unlikely classes and boxes).

$$SG_{init} = \underset{B \subset U, |B|=k}{\text{argmax}} \sum_{(s^v, o^v, s^e, p^r, o^e) \in B} \Big(\log P_{det}^{(0)}[s^v, s^e] + \log P_{rel}[s^v, o^v, p^r] + \log P_{det}^{(0)}[o^v, o^e] \Big) \quad (27)$$

8.2.4 Iterative Detection Scores Estimation

Careful readers may notice the superscript $^{(0)}$ in grounding vector $\boldsymbol{g}^{(0)}$, attention $A^{(0)}$, instance label $Y_{det}^{(0)}$, and instance prediction $P_{det}^{(0)}$. We use the superscript $^{(0)}$ to denote these are initial grounding results, which could be improved by the WSOD iterative refining technique proposed in [248]. Suppose loss $L_{det}^{(t)}$ $(t \geq 0)$ brings $P_{det}^{(t)} \in \mathbb{R}^{n_v \times c_e}$ close to $Y_{det}^{(t)} \in \mathbb{R}^{n_v \times c_e}$, where $Y_{det}^{(t)}$ is the caption-guided target label and $P_{det}^{(t)}$ is the prediction without help from captions. We could then incorporate the entity information $E = [e_1 \cdots e_{n_e}]^T$ of the caption into $P_{det}^{(t)}$ to turn it into a stronger instance-level label $Y_{det}^{(t+1)}$. The motivation is that the initial label $Y_{det}^{(0)}$ extracted from attention (Eq. 22, 25) will be easily influenced by the *noise in captions*. Since the attention scores always sum to one, some region will be assigned a higher score than others, regardless of whether the objects have consistent visual appearance. In an extreme case, mentioned but not visually present entities also have a matched proposal. Using $P_{det}^{(t)}$ is an indirect way to also consider the visual model's (Eq. 26) output, which encodes the objects' consistent appearance.

To turn $P_{det}^{(t)}$ into $Y_{det}^{(t+1)}$, we first extract $A^{(t+1)} \in \mathbb{R}^{n_e \times n_v}$ (same shape as the attention matrix $A^{(0)}$). We simply select the columns (denoted as $[:, i]$) from $P_{det}^{(t)}$ according to E to achieve $A^{(t+1)}$, and compute $\boldsymbol{g}^{(t+1)}$ and $Y_{det}^{(t+1)}$.

$$\begin{aligned}
A^{(t+1)} &= \left[P_{det}^{(t)}[:, e_1] \cdots P_{det}^{(t)}[:, e_{n_e}] \right]^T \\
\boldsymbol{g}^{(t+1)} &= \operatorname*{argmax}_{j \in \{1 \cdots n_v\}} A^{(t+1)}[i, j] \qquad (28) \\
Y_{det}^{(t+1)}[i, j] &= 1 \text{ if } \exists k \in \{1 \cdots n_e\}, s.t.(\boldsymbol{g}_k^{(t+1)} = i, e_k = j)
\end{aligned}$$

We refine the model n_t times, and in Eq. 25, we use $\boldsymbol{g}^{(n_t)}$ from the last iteration to compute Y_{relsub} and Y_{relobj}.

8.2.5 Modeling Sequential Context

We observed the model sometimes generates triplets that violate common sense, e.g., plate-on-pizza in Fig. 35 top, because the aforementioned test time post-processing (Eq. 27) considers predictions from P_{det} and P_{rel} separately. When joined, the results may not form a meaningful triplet. To solve the problem, we propose a vision-language module to consider sequential patterns summarized from the dataset (Fig. 32 middle-bottom). The idea is inspired by [151], but different because: (1) we encode the language and vision priors within the same multi-modal RNN while [151] models vision and language separately, and (2) our label generation captures a language N-gram such that the later generated object and predicate will not contradict the subject.

Specifically, we gather the grounded tuples $D_{gt} = \{(v_{\boldsymbol{g}_{s_i}}, v_{\boldsymbol{g}_{o_i}}, e_{s_i}, r_i, e_{o_i})\}_{i=1}^{n_r}$ within each training example to learn the sequential patterns. Compared to the SGGen 5-tuple (Eq. 27), the e_{s_i}, r_i, e_{o_i} here are from the ground-truth (E, R) and are always correct (e.g., no "cake-eat-person"). Since the module receives high-quality supervision from captions, it will assign low scores or adjust the prediction (Eq. 27) for imprecise 5-tuples at test time, using its estimate of what proper 5-tuples look like.

Fig. 32 middle-bottom shows the idea. We use an RNN (LSTM in our implementation) to consume both word embeddings and visual features of the subject and object. The training outputs are subject prediction $P_{cssub} \in \mathbb{R}^{n_r \times c_e}$ ($_{cs}$ for *c*ommon *s*ense), object prediction $P_{csobj} \in \mathbb{R}^{n_r \times c_e}$, and predicate prediction $P_{cspred} \in \mathbb{R}^{n_r \times c_r}$. We now explain how to generate their i-th row (to match true e_{s_i}-r_i-e_{o_i}).

First, we feed into the RNN a dummy */start/* embedding and the grounded subject visual feature $\boldsymbol{v}_{\boldsymbol{g}_{s_i}}$. The subject prediction $P_{cssub}[i,:]$ is achieved by a linear layer projection (from RNN output to d-dim) and matrix multiplication (using $W_{ent} \in \mathbb{R}^{c_e \times d}$). We predict the object $P_{csobj}[i,:]$ similarly, but using the grounded object visual feature $\boldsymbol{v}_{\boldsymbol{g}_{o_i}}$ concatenated with the subject word embedding e_{s_i} as inputs. If we do not consider the visual input, this step is akin to learning a subject-object 2-gram language model. Next, the RNN predicts predicate label $P_{cspred}[i,:]$ (using $W_{rel} \in \mathbb{R}^{c_r \times d}$ instead of W_{ent}), using object word embedding e_{o_i} and a dummy visual feature $\emptyset$ as inputs.

To learn $P_{cssub}, P_{csobj}, P_{cspred}$, we extract labels $Y_{cssub}, Y_{csobj}, Y_{cspred}$ (Eq. 29) and use cross-entropy losses $L_{cssub}(P_{cssub} \sim Y_{cssub})$, $L_{csobj}(P_{csobj} \sim Y_{csobj})$, $L_{cspred}(P_{cspred} \sim Y_{cspred})$ to optimize the RNN model.

$$\begin{aligned} Y_{cssub} &= \Big[Y_{ent}[e_{s_i},:]^T \cdots Y_{ent}[e_{s_{n_r}},:]^T\Big]^T \\ Y_{csobj} &= \Big[Y_{ent}[e_{o_i},:]^T \cdots Y_{ent}[e_{o_{n_r}},:]^T\Big]^T \\ Y_{cspred} &= Y_{rel} \end{aligned} \tag{29}$$

At test time, we feed to the RNN the visual features from SG_{init} (Eq. 27) and the */start/* embedding. We let the RNN re-label the subject-object-predicate using beam search. The final score for each re-labeled 5-tuple is the sum of log probabilities of generating subject, object, and predicate. We generate the object before the predicate because objects are usually more distinguishable than predicates, so this order simplifies inference, allowing the use of a smaller beam size. We re-rank the beam search results using the final scores and keep the top ones to compute the Recall@k to evaluate (examples in Fig. 35, Fig. 36).

Our final model is trained using the following multi-task loss, where β is set to 0.5 since at the core of the task is the grounding of visual objects.

$$L = L_{grd} + \beta\big(\sum_{t=0}^{n_t} L_{det}^{(t)} + L_{relsub} + L_{relobj} + L_{cssub} + L_{csobj} + L_{cspred}\big) \tag{30}$$

8.3 Experiments

Datasets. We use the Visual Genome (VG) [126] and Common Objects in Context (COCO) [147] datasets, which both provide captions describing the visual contents. VG involves 108,077 images and 5.4 million region descriptions. The associated annotations of 3.8 million object instances and 2.3 million relationships enable us to evaluate the scene graph generation performance. To fairly compare to the counterpart weakly-supervised scene graph generation methods [317, 313], we adopt the VG split used in Zareian *et al.* [313]: keeping the most frequent $c_e = 200$ entity classes and $c_r = 100$ predicate classes, resulting in 99,646 images with *subject-predicate-object* annotations. We use the same 73,791/25,855 train/test split[1]. We also adopt the split in Xu *et al.* [284], more commonly used by fully-supervised methods. It contains 75,651/32,422 train/test images and keeps $c_e = 150$ entity and $c_r = 50$ predicate classes. Both VG splits are preprocessed by [313].

For COCO data, we use the 2017 training split (118,287 images). We rule out the duplicated images in the VG test set, resulting in 106,401 images for Zareian *et al.*'s split and 102,786 images for Xu *et al.*'s.

Learning tasks. The linguistic structure supervision for training is from the following three sources:

- VG-GT-Graph imagines an ideal scenario (an upper bound with the noise in captions and parsers' impacts isolated) where we have the ground-truth text graph annotations instead of a set of image-level *subject-predicate-object* triplets, for training on VG. To get these ground-truth graphs, we check the visual regions associated with the entities (subjects

[1] We follow [313], but [317] reports 73,801/25,857 train/test split

and objects) and connect entities if their regions have IoU greater than 0.5. We do *not* use box annotations to improve detection results.

- VG-Cap-Graph utilizes the VG *region* descriptions. We use [219] to extract text graphs from these descriptions, but we ignore the region coordinates and treat the graphs as image-level annotations.
- COCO-Cap-Graph uses captions from COCO and applies the same parsing technique as VG-Cap-Graph. The difference is that these captions are image-level, and describe the objects and relations as a whole.

Metrics. We measure how accurately the models generate scene graphs, using the densely-annotated scene graphs in the VG test set. Following [284], a predicted triplet is considered correct if the three text labels are correct and the boxes for subject/object have ≥ 0.5 IoU with ground-truth boxes. We then compute the Recall@50 and Recall@100 as the fraction of the ground-truth triplets that are successfully retrieved in the top-50 and top-100 predictions, respectively.

Methods compared. We conduct ablation studies to verify the benefit of each component of our method.

- Basic model refers to our Sec. 8.2.2-8.2.3 without applying the phrasal contextualization. We set $\psi(E, G^L) = H_{ent}^{(0)}$.
- +Phrasal context (Sec. 8.2.1) uses contextualized entity embeddings $\psi(E, G^L)$ instead of $H_{ent}^{(0)}$.
- +Iterative (Sec. 8.2.4) gradually improves the grounding vector $\boldsymbol{g}$. We iterate $n_t = 3$ times by default.
- +Sequential context (Sec. 8.2.5) revises the prediction presented in Eq. 27, using the RNN encoded with knowledge regarding sequential patterns.

We compare to weakly-supervised scene graph generation methods that published results on Zareian *et al.*'s split: VtransE-MIL [316], PPR-FCN-single [317], PPR-FCN [317] and VSPNet [313]. We also compare to fully-supervised methods on Xu *et al.*'s split: Iterative Message Passing (IMP) [284], Neural Motif Network (MotifNet) [315], Associative Embedding (Asso.Emb.) [187], Multi-level Scene Description Network (MSDN) [141], Graph R-CNN [289], and fully-supervised VSPNet [313].

8.3.1 Results on GT-Graph Setting

The GT-Graph setting allows our method to be fairly compared to the state-of-the-art methods because in this setting, the information ours and those methods receive is comparable (sets of triplets, in our case connected). Further, the word distribution is the same for training/testing, while the caption setting causes a train-test shift (described shortly).

Table 36: SGGen recall (%) under VG-GT-Graph setting. We compare our method to the state-of-the-art methods. High recall (R@50, R@100) is good.

Zareian *et al.*'s split (weakly sup)			Xu *et al.*'s split (fully sup)		
Method	R@50	R@100	Method	R@50	R@100
			IMP [284]	3.44	4.24
VtranE-MIL [316]	0.71	0.90	MotifNet [315]	6.90	9.10
PPR-FCN-single [317]	1.08	1.63	Asso.Emb. [187]	9.70	11.30
PPR-FCN [317]	1.52	1.90	MSDN [141]	10.72	14.22
VSPNet [313]	3.10	3.50	Graph R-CNN [289]	11.40	13.70
			VSPNet (Full) [313]	12.60	14.20
Basic	2.20	2.88	Basic	3.82	4.96
+ Phrasal	2.77	3.62	+ Phrasal	4.04	5.21
+ Iterative	3.26	4.15	+ Iterative	6.06	7.60
+ Sequential	4.92	5.84	+ Sequential	7.30	8.73

In Tab. 36 left, we show our results on Zareian *et al.*'s VG split and baselines of weakly-supervised methods. Our Basic method already surpasses VtransE-MIL, PPR-FCN-single, and PPR-FCN. This may be due to the low quality of the EdgeBox proposals used in them. Compared to VSPNet, which also uses Faster RCNN proposals, our Basic method is slightly worse, but our components greatly improve upon Basic, and our final model achieves 4.92, a 59% improvement over VSPNet (using R@50). +Phrasal context improves Basic by 26% (2.77 v.s. 2.20), +Iterative improves +Phrasal by 18% (3.26 v.s. 2.77), and +Sequential gains 51% (4.92 v.s. 3.26).

In Tab. 36 right, we compare to fully-supervised methods on Xu *et al.*'s split. We observe our method is very competitive even though we only use image-level annotations. In terms of Recall@50, our final method (7.30) outperforms IMP (3.44) and MotifNet (6.90). As for the relative improvement, +PHRASAL context improves BASIC by 6% (4.04 v.s. 3.82), +ITERATIVE gains 50% (6.06 v.s. 4.04), and +SEQUENTIAL gains 20% (7.30 v.s. 6.06).

8.3.2 Results on Cap-Graph Setting

Our proposed Cap-Graph setting is an under-explored and challenging one, as the learned SGGen model depends on the captions' exhaustiveness and the parser's quality, but it allows learning from less expensive image-text data.

Table 37: SGGen recall (%) under Cap-Graph settings. High recall (R@50, R@100) is good.

	VG-Cap-Graph				COCO-Cap-Graph			
Eval split	Zareian *et al.*'s		Xu *et al.*'s		Zareian *et al.*'s		Xu *et al.*'s	
	R@50	R@100	R@50	R@100	R@50	R@100	R@50	R@100
BASIC	0.81	0.91	0.99	1.09	1.20	1.51	2.09	2.63
+ PHRASAL	0.90	1.04	1.39	1.69	1.17	1.47	1.65	2.16
+ ITERATIVE	1.11	1.32	1.79	2.22	1.41	1.75	2.41	3.02
+ SEQUENTIAL	1.83	1.94	3.85	4.04	1.95	2.23	3.28	3.69

In Tab. 37, we show the SGGen performance of models learned from VG region captions (VG-Cap-Graph) and COCO image captions (COCO-Cap-Graph). We see the same trend as in GT-Graph setting: our components (+PHRASAL, +ITERATIVE, and +SEQUENTIAL) have positive effects. Further, our final models learned from both VG-Cap-Graph (R@50 1.83) and COCO-Cap-Graph (1.95) are better than all weakly-supervised methods except VSPNet (in Tab. 36 left). Our models learned from captions are even comparable (VG-Cap-Graph 3.85, COCO-Cap-Graph 3.28) to the fully-supervised IMP (R@50 3.44).

Fig. 33 shows the relation frequencies in our settings. We observe that some relations ("has," "near") rarely appear in text descriptions but are often annotated in ground-truth scene graphs. Meanwhile, there are frequently mentioned prepositions in captions ("of," "with," "at") which are rarely denoted as relations. These train/test discrepancies (train on captions, test on triplets) explain our methods' relative performance in Tab. 37, where +PHRASAL helps under VG-Gap-Graph (more similar to GT-Graph) but hurts slightly under COCO-Cap-Graph (less similar to GT-Graph).

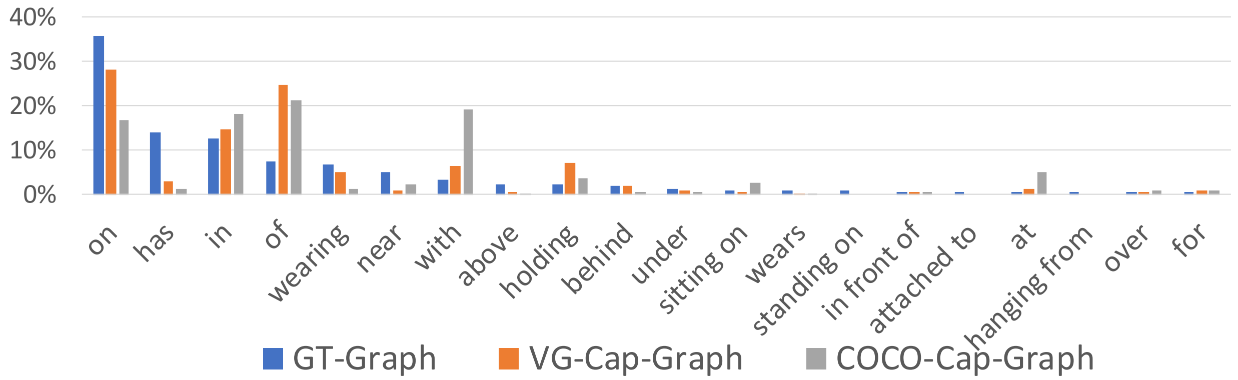
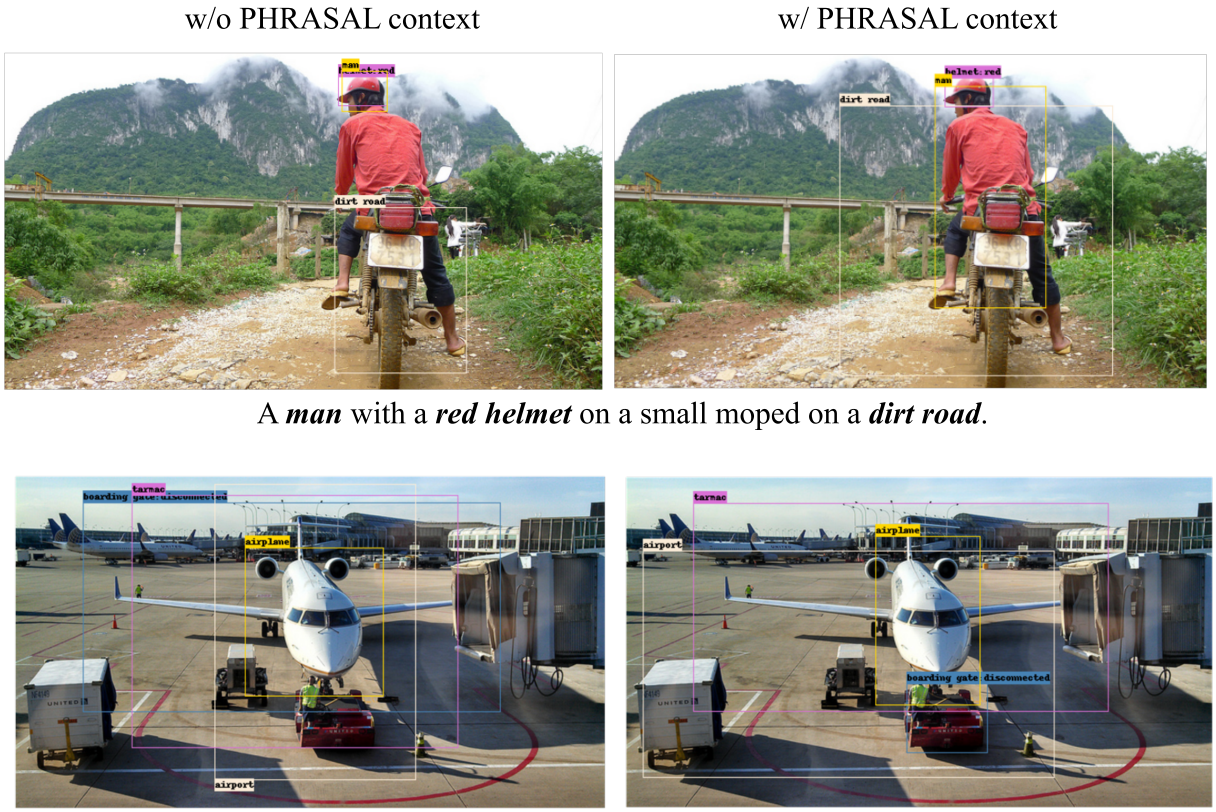


Figure 33: Relation frequencies in the three settings.

8.3.3 Qualitative Examples

Fig. 34 compares using and not using PHRASAL context. Without out PHRASAL module (left), the grounding procedure gets stuck on the same distinguishable local region (top-left: head of man) or erroneously attends to the whole image (bottom-left: boarding gate). When using the PHRASAL module (right), our model is better at localizing visual objects. It knows there should be a complete person in the scene (top-right) and the boarding gate is a concept related to the plane (bottom-right).

Fig. 35 shows how the learned sequential patterns help correct imprecise predictions. For the corrections (beam size=5), we show the log-probability of the 5-tuple and individual probabilities. Given that *plate* cannot be put *on pizza*, our model corrects it to *plate-under-pizza*. In the bottom example, our model corrects *person-wear-person* to *person-wear-shirt* and *person-behind-person*. In Fig. 36, we compare our BASIC and final methods.



A ***man*** with a ***red helmet*** on a small moped on a ***dirt road***.

An ***airplane*** sits on the ***tarmac*** of an ***airport***, with a ***disconnected boarding gate***.

Figure 34: Importance of PHRASAL context; best seen with zoom.

8.3.4 Implementation Details

We pre-extract text graphs using [282]'s implementation of [219]. We use the same proposals ($n_v = 20$ per image) and features ($d_{cnn} = 1536$) as [313], extracted using Faster-RCNN [211] (InceptionResnet backbone [243]) pre-trained on OpenImage [131]. During training, we use GraphNets [17] to encode phrasal context. W_{ent}, W_{rel} are $d = 300$ frozen GloVe embeddings [200]. To train our model, we use a batch size of 32, learning rate 0.00001, the Adam optimizer [119], and Tensorflow distributed training [1]. We use weight decay of 1e-6 and the random normal initializer (mean=0.0, stdev=0.01) for all fully-connected layers. We use LSTM cell, 100 hidden units, and dropout 0.2, for the SEQUENTIAL module. For the non-max-suppression of Eq. 27, we use score threshold 0.01, IoU threshold 0.4, and limit the maximum instances per entity class to 4. We set beam size to 5 for the SEQUENTIAL

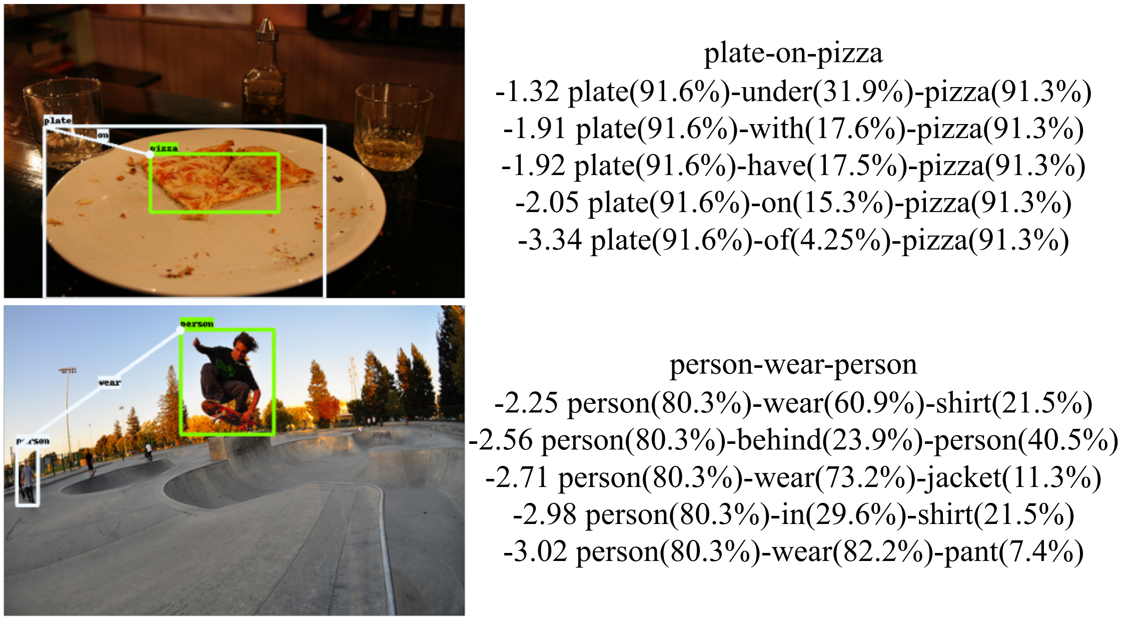


plate
on
pizza
plate-on-pizza
-1.32 plate(91.6%)-under(31.9%)-pizza(91.3%)
-1.91 plate(91.6%)-with(17.6%)-pizza(91.3%)
-1.92 plate(91.6%)-have(17.5%)-pizza(91.3%)
-2.05 plate(91.6%)-on(15.3%)-pizza(91.3%)
-3.34 plate(91.6%)-of(4.25%)-pizza(91.3%)
person
wear
person
person-wear-person
-2.25 person(80.3%)-wear(60.9%)-shirt(21.5%)
-2.56 person(80.3%)-behind(23.9%)-person(40.5%)
-2.71 person(80.3%)-wear(73.2%)-jacket(11.3%)
-2.98 person(80.3%)-in(29.6%)-shirt(21.5%)
-3.02 person(80.3%)-wear(82.2%)-pant(7.4%)

Figure 35: Importance of SEQUENTIAL context.

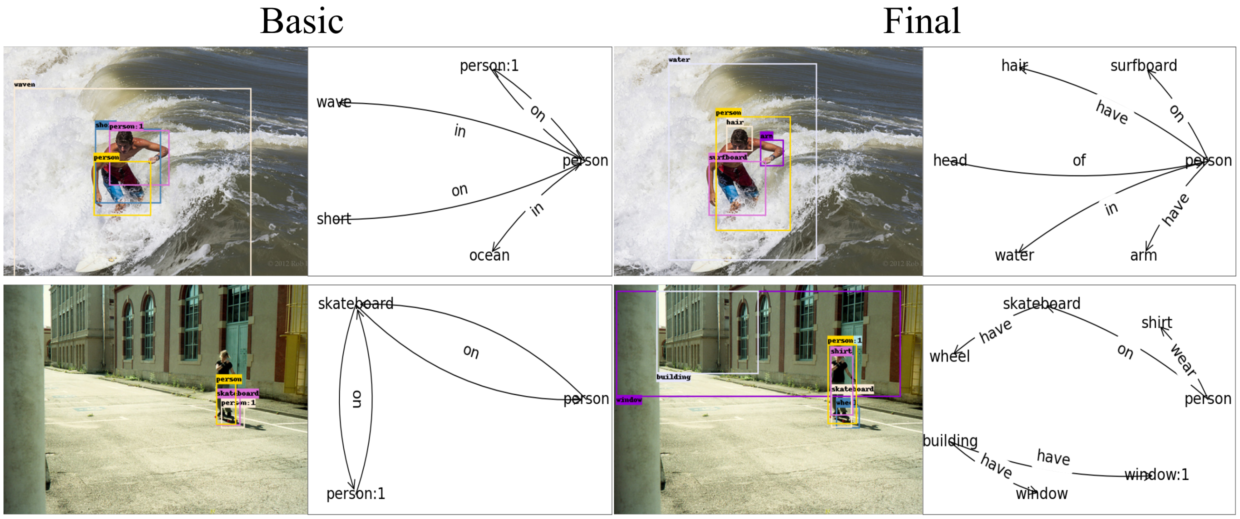


Figure 36: BASIC v.s. our final model; best viewed with zoom.

module post-processing.

8.4 Conclusion

In this chapter, we proved the thesis hypotheses H3 and H4 (see Tab. 38). Our proposed weakly supervised scene graph generation is a natural extension of Chapter 7 since it further gouges the information beyond entities from the caption. We proposed to use both the phrasal and sequential contexts. The former captures the entities shared in different subject-predicate-object triplets while the latter captures language nature. The two components together pushed forward the limit of utilizing the information from captions, and increased the reliability of our weakly supervised SGGen model.

To isolate the contribution of global context from the noise contained in captions, we verify our approach in two settings. We construct a ground-truth triplet graph by connecting triplets with certain overlap. We show that our full method greatly outperforms prior work (it boosts the performance of [313] by 59%-67%). Second, we use two types of actual captions. We observe that modeling phrasal and sequential linguistic context achieves strong results, significantly better than more direct uses of captions, and competitive with methods using clean image-level supervision.

Table 38: Conclusion - validated hypotheses in this chapter.

	Multimodal features help to understand images/videos with implicit persuasive intent, such as visual advertisements.	Text features can be unreliable if not modeled appropriately.	Text supervision contains noise, but can be used to localize visual objects in space, if modeled properly.	Text supervision provides contexts regarding visual objects, they are reliable cues for disambiguating entities and relations.	Noisy audio narrations as a multimodal signal can be modeled to localize video actions in temporal domain.
This Chapter			✓	✓	

9.0 Action Detection through Audio Narration Supervision

9.1 Introduction

The last two chapters discussed the way of using captions as supervised signals to learn image detection models: Chapter 7 proposed a way to choose the proper image-caption pairs for training and designed a method to learn detector from the selected data; Chapter 8 further analyzed the linguistic information from the texts and additionally built the relationships among detected objects. At the core of the two is our exploration of using captions as a novel and cheap even if potentially noisy supervised signal.

Inexpensive and informative annotations are also needed in the video domain. However, it is challenging to explore such a signal to achieve the goal of localizing specific actions. Videos involve natural annotations such as video titles, narrations, and even sparse frame-level data. Meanwhile, it also features the multi-modalities such as RGB frames, motion features, and ambient sound. These complicated factors bewilder researchers because not every single feature provides hints regarding localization. Besides, adopting the signal to predict instance-level detection results requires proper modeling, which is also not simple.

This chapter will explore audio narrations in the untrimmed video action detection task. We first distinguish the narration annotations from the instance-level or video-level annotations. We show in Fig. 37 an example video clip from the EPIC Kitchens dataset [45], as well as the different forms of supervisions. The instance-level annotations are defined by triplets (start time, end time, action class). Models trained under the fully-supervised setting can use this form of supervision to generte temporal action proposals [146, 145], or process with action detection [57, 71, 157, 231, 236, 300, 322]. The major benefit of instance-level data is that the result model are usually boundary-sensitive, thus the foreground and background are clearly distinguished by the detection scores, causing high average precision.

However, fully annotating a video dataset with instance-level labels is even more time-consuming than annotating the bounding boxes for the image object detection tasks. Thus, methods [129, 190, 191, 197, 230, 269] focus on the weakly supervised action detection

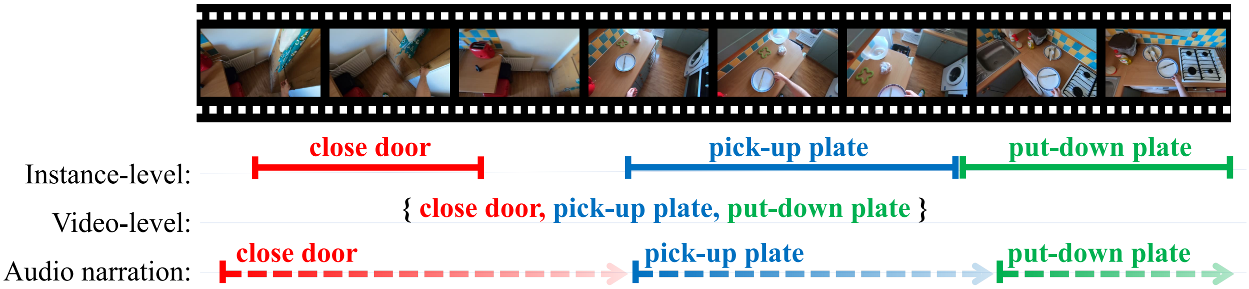


Figure 37: Instance-level, video-level, and audio narration supervisions. The audio narration supervision in the EPIC Kitchens dataset only includes a imprecise start time.

(WSAD), which only requires video-level labels. These methods assume the video to be a bag of actions and use multi-label cross-entropy loss to optimize. One disadvantage of WSAD is that it assumes only a few classes per video (e.g., < 5, see Tab. 39). Hence, it is not applicable in real cases. In an extreme scenario, a 2-hour untrimmed video may consist all of the action classes, thus the video-level label is too coarse for the model to learn a good detector.

Table 39: Datasets information. Most WSAD methods use THUMOS 14 [95], in which there is only 1 action class per video. We explore single-timestamp audio narration annotations in EPIC Kichens [26].

Dataset	Avg. video length (secs)	Avg. classes per video	Avg. actions per video
THUMOS 14	209	1.08	15.01
EPIC Kitchens	477	34.87	89.36

We shall use the audio narrations from EPIC Kitchens [45] dataset. Different from the ambient sound track, videos were narrated by the annotators to gather an extra narration audio track. The recorded narration track are later transcribed into texts, and then parsed into action classes (verb + noun) using a dependency parser, resulting in the forms we use (see Fig. 37 bottom).

The major challenge of using audio narrations is that the ***annotations are noisy***. Unlike the instance-level annotation: 1) narration annotations' start timestamps are not precise in that they may overlap with the previous action, and 2) the end time is unknown since the narrators provide no hints regarding the end time. One can assume the end time to be before the start time of the next action, but the frames in between are in a gray period, and the belonging is unsure. Therefore, the narrations provide a trade-off between accurate instance-level annotation and cheap and fast video-level annotation. One needs to model the uncertainty to use them.

To use the narrations to learn action detection model, we first cut the untrimmed videos into clips using the single timestamp (start time) of the audio narration annotations (see Fig. 37). Thus, each clip can be treated a mixture of actions given that the boundaries are imprecise. Then, the association between the frames in the clip and the clip-level action class could be solved by the Multiple Instance Learning (MIL). As compared to the common WSAD methods such as [190, 191] which only distinguish between foreground and background (see statistics of THUMOS 14 in Tab. 39), the background in our clip may involve other semantically meaningful actions. So, we design a class-aware attention mechanism to assign higher scores to the frames in the clip that are more related to the narrated class. Meanwhile, we extract video multimodal features from RGB frames, motion flow, and ambient sound. We apply a simple early fusion architecture to the problem and ablate the contributions of each modality.

To summarize, our contributions are as follows:

- We propose to use the audio narrations to learn video action detection model. To our best knowledge, this is a brand new task that had not been explored (in EPIC Kitchen tasks C1-C5, only C1-weakly is marginally related but different than our task because C1-weakly requires only to classify trimmed video at test time while ours requires to localize actions in untrimmed videos).
- We provide a solution to the proposed task, in which we use class-aware attention mechanism to rule-out video frames that are not related to the narration label. Also, our solution considers multimodal features including RGB frames, motion flow feature, and audio.

- We ablate our method on the EPIC-Kitchen dataset, and analyze the contributions of each model component and feature modality.

9.2 Approach

We first formulate the audio narration guided WSAD task and then overview the model training pipeline. Then, we introduce the details regarding the multimodal features in Sec. 9.2.1, discuss the design of the proposed class-aware attention in Sec. 9.2.2, and provide the post-processing algorithm which turns the frame-level prediction into instance-level (required by evaluation), in Sec. 9.2.3.

Task formulation: At training time, the video and the paired $\{time_i, verb_i, noun_i\}_{i=1}^{N}$ as N annotated actions are provided, where $time_i$ is the narration start time, $verb_i$ and $noun_i$ are the narrated verb and noun classes accordingly. Note that here the underlying assumption is that the $time_i$ is not precise to represent the narration starting time since there may be overlap between consecutive actions. At test time, models have to predict four tuples of $\{time\_s_i, time\_e_i, verb_i, noun_i\}$ given the video, where $time\_s_i, time\_e_i$ are the starting and ending time accordingly.

Training pipeline overview(Fig. 38): Given a video and the paired audio narration annotation $\{time_i, verb_i, noun_i\}_{i=1}^{N}$, we first split the video into training clips. Given a specific action $(time_i, verb_i, noun_i), i \in \{1 \dots N\}$, we cut the video from $time_i$ to $time_{i+1}$, resulting in a video clip (N clips in total) paired with $verb_i$ and $noun_i$. We denote the frames in the i-th clip as $\{f_{i,j}\}_{j=1}^{L_i}$ where L_i is the total number of frames in the i-th video clip.

Then (Fig. 38 (middle)), we proceed with the feature extraction process, which will be explained in detail in Sec. 9.2.1. Briefly, we extract the visual CNN features of both the RGB and Flow frames, and the semantic embedding of the ambient soundtrack. After feature extraction, we use early fusion to aggregate these multiple modalities.

Finally, we use an audio narration class guided attention mechanism to filter out irrelevant classes in the clip, given that the i-th clip should be all regarding the $verb_i$ and $noun_i$. For example, in Fig. 38 (top), we show that given the verb class "put-down" as the query, the attention will focus on the last few frames (potentially have overlap with the next action). However, given the narration supervision of the clip, we need to choose the attention distribution of the verb "pick-up" among frames. The frames that highlighted by the attention scores are then responsible for predicting the clip-level action (e.g., "pick-up").

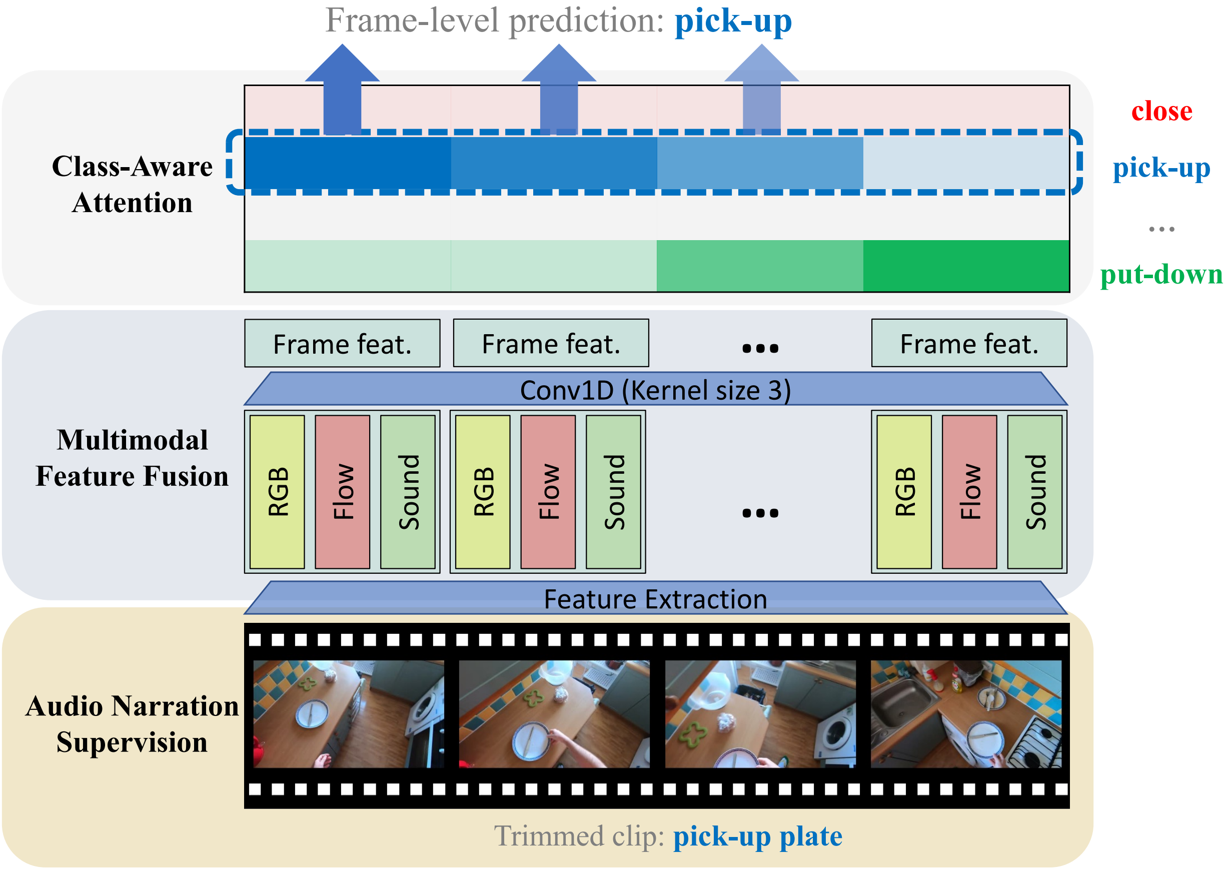


Figure 38: Model overview. We first cut the video into clips using the single timestamp denoted in the audio narration. Then, each video clip can be treated as a bag of a few actions. Next, we extract multimodal features and use early fusion to combine them (Sec. 9.2.1). We use a class-aware attention mechanism to produce the frame-level detection score (Sec. 9.2.2). Finally, we use a class-aware, intensity-sensitive post-processing (Sec. 9.2.3) to turn the frame-level prediction into instance-level (not shown), for evaluation purpose.

9.2.1 Multimodal Video Features

We consider features from the following sources. Though using them seems to be common in video recognition, we are working on a novel task of learning action detection models from narration supervision, in which the contributions of RGB/flow/audio features are unclear.

- **RGB and flow frames.** We use the standard RGB and flow features provided in the EPIC Kitchens dataset [45], i.e., the 1024-D RGB and 1024-D Flow CNN features generated by a TSN model[63] pre-trained in [45].
- **Ambient sound.** Since the EPIC Kitchens dataset provides the soundtrack of the ambient audio, we also model them because sound may imply some actions. We use the VGGish [68] to produce a 128-D semantic embedding for every second. The VGGish method was first used in the AudioSet [68] classification task and it was pre-trained on a large YouTube dataset (later became YouTube-8M).

Early fusion of the multimodal video features. We linearly interpolate the ambient sound semantic embeddings, to convert its sequence lengths to be the same as the RGB and flow features. We denote the concatenation of these multimodal features as the video frame feature $\{\boldsymbol{f}_{i,j}\}_{j=1}^{L_i}$, $\boldsymbol{f}_{i,j} \in \mathbb{R}^{2176\times 1}$ (RGB 1024-D, Flow 1024-D, Ambient sound 128-D, L_i - number of frames). Let $\boldsymbol{F}_i = [\boldsymbol{f}_{i,1}\ \boldsymbol{f}_{i,2} \cdots \boldsymbol{f}_{i,L_i}]^T \in \mathbb{R}^{L_i \times 2182}$ be the video sequence feature, we apply a Conv1D layer (with kernel size 3, ReLu activation) to further extract the frame feature $\mathbf{F}_i = [\mathbf{f}_{i,1}\ \mathbf{f}_{i,2} \cdots \mathbf{f}_{i,L_i}]^T \in \mathbb{R}^{L_i \times d}$ ($d = 100$ is the number of neuron units):

$$\mathbf{F}_i = \text{Conv1D}(\boldsymbol{F}_i) \tag{31}$$

9.2.2 Class-Aware Attention for Weakly Supervised Action Detection

After getting the multimodal video features, we design a class-aware attention mechanism to localize the actions in the sequences. Our model selects relevant frames best represent the action in the video clip and uses their aggregated features to represent the video clip. Take Fig. 38 as an example, the verb class for the video clip is “pick-up” (i.e., clip-level label), so we use the embedding of “pick-up” to multiply (dot-product) each frame feature $\mathbf{f}_i$ to measure the frame-label similarity, resulting in a sequence of scores. After normalization, this score array represent the likelihood that the associated frames involve the action "pick-up". We compute the weighted sum of the sequence features (weighed by the normalized scores). Then, we add a classification layer to predict the action and use cross entropy loss to optimize.

Formally, we define action label embedding weights $\mathbf{W}^{(1)}_{verb} \in \mathbb{R}^{C_{verb} \times d}$, $\mathbf{W}^{(1)}_{noun} \in \mathbb{R}^{C_{noun} \times d}$ where C_{verb} and C_{noun} are the number of verb and noun classes, respectively. Since the verb detection and noun detection follow the same pipeline and only differ in the number of classes, we use $\mathbf{W}^{(1)} = \mathbf{W}^{(1)}_{verb}$ or $\mathbf{W}^{(1)}_{noun}$ as an abstract notation to denote the either label embedding, $C = C_{verb}$ or C_{noun} to denote the number of classes, and $c_i = verb_i$ or $noun_i$ to denote the clip-level label. Then, the following procedure applies to both verb and noun detection parallel tasks.

We first compute the dot-product between the label embedding and the frame feature, then, we use the sigmoid function to turn the score into a probability $\mathbf{A}'_i \in \mathbb{R}^{C \times L_i}$ (see Eq. 32; Fig. 38 (top) shows $\mathbf{A}'_i$ using the color matrix). Since we are aware of the class that is narrated in the video clip, we select the specific c_i-th row in $\mathbf{A}'_i$ ($c_i = verb_i$ or $noun_i$), resulting in $\mathbf{A}_i \in \mathbb{R}^{1 \times L_i}$. This class-aware row selection process is shown in Fig. 38 (top) using the blue dashed box.

$$\mathbf{A}'_i = \text{sigmoid}(\mathbf{W}^{(1)}\ \mathbf{F}_i^T), \qquad \mathbf{A}_i = \mathbf{A}'_i[c, :] \tag{32}$$

Meanwhile, we use a fully connected layer $\mathbf{W}^{(2)} \in \mathbb{R}^{d \times C}$ to estimate the per-frame detection score $\mathbf{D}_i \in \mathbb{R}^{L_i \times C}$. In Eq. 33, $j \in \{1 \cdots L_i\}$ denotes frame id and $k \in \{1 \cdots C\}$ is the class index.

$$\mathbf{D}'_i = \mathbf{F}_i\ \mathbf{W}^{(2)}, \qquad \mathbf{D}_i[j, k] = \frac{\exp\left(\mathbf{D}'_i[j, k]\right)}{\sum_{k'=1}^{C} \exp\left(\mathbf{D}'_i[j, k']\right)} \tag{33}$$

Directly optimizing the per-frame detection score $\mathbf{D}_i = \mathbf{D}_{verb\ i}$ or $\mathbf{D}_{noun\ i}$ is hard since we only have the clip-level label $c_i = verb_i$ or $noun_i$. Thus, we apply the class-aware attention weighting $\mathbf{A}_i$ to aggregate frame-level information into $\bar{\mathbf{F}}_i \in \mathbb{R}^{1 \times d}$ (Eq. 34), which is a clip-level feature. Then, the clip-level prediction is given by $\mathbf{P}_i \in \mathbb{R}^{C \times 1}$ (Eq. 35), which shares the $\mathbf{W}^{(2)}$ with Eq. 33.

$$\bar{\mathbf{F}}_i = \frac{\mathbf{A}_i \mathbf{F}_i}{\sum_{j=1}^{L_i} \mathbf{A}_i[j]} \tag{34}$$

$$\mathbf{P}'_i = (\bar{\mathbf{F}}_i\ \mathbf{W}^{(2)})^T, \qquad \mathbf{P}_i[k] = \frac{\exp\left(\mathbf{P}'_i[k]\right)}{\sum_{k'=1}^{C} \exp\left(\mathbf{P}'_i[k']\right)} \tag{35}$$

Finally, we use cross-entropy to optimize the model, where $\boldsymbol{y}_i$ is the one-hot representation of c_i ($\boldsymbol{y}_i[k] = 1$ iff $k = c_i$).

$$L = -\sum_i \sum_{k=1}^{C} \boldsymbol{y}_i[k] \log \mathbf{P}_i[k] \tag{36}$$

9.2.3 Class-Aware Intensity-Sensitive Post Processing

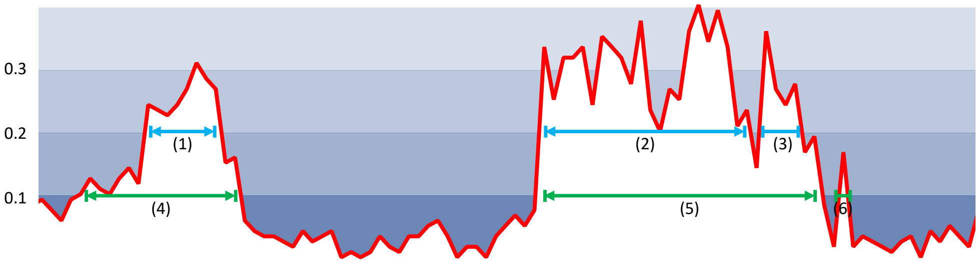


Figure 39: Intensity-sensitive post-processing. For each of the action classes, we use a set of thresholds (e.g., {0.1, 0.2}) and retrieve all the segments (consecutive frames) that meet the different threshold conditions. The retrieval results are a bunch of action segments with different intensities. Next, we score each segment and apply Non-Maximum Suppression (NMS) to remove highly overlapped (measured by IoU) detections. We show the action clips detected using a threshold of 0.1 using green color and the clips detected by threshold 0.2 using blue. Assuming the IoU threshold of 0.6, segment (5) will be removed because it highly overlapped with (2).

To get the detections in the form of $\{time\_s_i, time\_e_i, verb_i, noun_i\}$ from the frame-level prediction $\mathbf{D}_i$ (Eq. 33), we use a class-aware intensity-sensitive post process. Specifically, we consider each action class separately. Given the detection score of a specific class (e.g., the

k-th class in verb detection $\mathbf{D}_{verb\ i}[:, k]$), we first use different thresholds to retrieve the segments, which are defined to be the longest sequence of consecutive frames that have detection scores past the threshold. The result is a bunch of potential action segments detected by different intensity scores (i.e., thresholds). We then assign a score to each segment, denoting the averaging detection intensity within the segment. In Fig. 39, we show the segments detected by threshold 0.1 and 0.2 using green and blue colors, respectively. In the next step, we use Non-Maximum Suppression (NMS) to remove highly overlapped (measured by IoU) detections and retain only those with higher intensity. Finally, we aggregate the NMS-ed detections from all action classes and sort them by intensity, resulting in our final detection results.

9.3 Experiments

We provide the details regarding our model in Sec. 9.3.1. Then, we provide experimental results in Sec. 9.3.2, including analysis regarding both the contributions of our model components and the benefits of multimodal features. To better understand our model, we also provide qualitative results in Sec. 9.3.3.

9.3.1 Implementation Details

Before training the detector, we offline extract the multimodal features. The CNN features of the RGB and Flow frames are from [45], while we pre-processed the audio features. We use FFMpeg to extract audios from MP4 videos and feed the Mel spectrogram to the VGGish [68] model pre-trained on the large Youtube dataset (latter becomes Youtube-8M) to produce semantic audio embedding. After getting the above features, we interpolate the audio features to make them the same lengths as the RGB and Flow features.

We concatenate the multimodal features as the model input and add a Conv1D layer (with $d = 100$ filters, kernel size 3, ReLu activation) to further finetune. During training, we use a dropout probability of 0.5 for the Conv1D layer, a dropout probability of 0.5 for the learned attention ($\mathbf{A}_i$). We use the Tensorflow framework [1], Adam optimizer [119], a learning rate of 1e-5, and a batch size of 8 (8 clips). All models in our experimental sections are trained for 300K steps on the EPIC kitchens dataset, using a validation set to pick the best model.

For the post-processing, we first apply uniform filtering (filter size 3) on each class's detection scores (e.g., $\mathbf{D}_{verb\ i}[:, k]$) to make the detection scores less fluctuated. Then, we vary the detection threshold from 0.01 to 0.4 to retrieve all segments and use NMS with an IoU threshold of 0.4 to remove highly overlapped segments.

9.3.2 Results on the EPIC Kitchens Dataset

Metrics. Although our training process does not rely on instance-level annotations, we can use the EPIC Kitchens' C2 task's (Action Detection) evaluation protocol, which measures the performance of the action detections. Basically, the protocol computes the average of the Average Precision (AP) values for each class, aka mean AP. A predicted segment is considered correct if its Intersection over Union (IoU) with a ground truth segment is greater than or equal to a given threshold (0.1 to 0.5). Besides the verb and noun detection, the EPIC Kitchens' C2 task also involves an action detection evaluation which requires the verb and noun detections to be correct at the same time.

Contributions of Proposed Components. We verify the effectiveness of the proposed model and compare to the fully- and weakly-supervised action detection methods:

- Ful. [44] is a fully supervised model trained by the EPIC Kitchens challenge organizer, using a two-stage approach to solve the action detection (action proposal [145] + action classification [57]).
- Our Ful. is a one-stage fully supervised method trained by us, in which we predict the frame-level actions then post-process (Sec. 9.2.3). We treat Our Ful. as a proper upper bound baseline in that all of our weakly supervised methods depend on similar frame-level prediction + post-processing.
- Narr. Bas. is the baseline method of using narration supervision. In Narr. Bas., we treat the single timestamp in the narration annotations as the boundaries and use the

cut result as instance-level annotations to directly train a fully supervised model.

- Cls. Agno. is an alternative method, in which we use a class agnostic attention instead of class-aware attention (Sec. 9.2.2).

Table 40: Contributions of proposed components. We show the Average Precision (%) at certain IoU thresholds (@0.1-@0.5) and the mean Average Precision (Avg.). All numbers are higher the better. The best weakly supervised model learned using narration annotations is shown in **bold** and the second best is in *italic*.

	Action Detection						Verb Detection						Noun Detection					
	@0.1	@0.2	@0.3	@0.4	@0.5	Avg.	@0.1	@0.2	@0.3	@0.4	@0.5	Avg.	@0.1	@0.2	@0.3	@0.4	@0.5	Avg.
Ful. [44]	6.95	6.10	5.22	4.36	3.43	5.21	10.8	9.84	8.43	7.11	5.58	8.36	10.3	8.33	6.17	4.47	3.35	6.53
Our Ful.	6.40	5.69	4.59	3.34	2.39	4.48	12.9	11.4	9.04	6.62	5.03	9.00	11.4	9.61	7.17	4.70	2.98	7.17
Narr. Bas.	4.42	3.62	2.91	2.06	1.47	2.90	9.39	7.45	5.68	3.99	2.85	5.87	8.43	**6.92**	**5.24**	**3.50**	**2.37**	**5.29**
Cls. Agno.	*4.57*	*3.78*	*3.10*	*2.28*	**1.70**	*3.09*	**10.0**	**8.53**	**7.03**	**4.79**	*3.40*	**6.75**	*8.49*	6.82	4.96	3.22	2.04	5.11
Ours	**4.68**	**4.01**	**3.27**	**2.33**	*1.65*	**3.19**	*9.64*	*7.96*	*6.31*	*4.70*	**3.56**	*6.43*	**8.51**	*6.88*	*5.09*	*3.36*	*2.25*	*5.22*

Tab. 40 shows the results. We found Our Ful., though a one-stage method, is very competitive to Ful. [44] (action detection mAP 4.48% v.s. 5.21%). The only weakness is that it is not that good at boundary refinement, hence its verb and noun detection AP@0.1,0.2,0.3 are higher but its AP@0.4,0.5 are lower. Then, Narr. Bas., which uses the same fully supervised method (but changes to use the narration supervision), inevitably hurts the action detection performance (action mAP 2.90% v.s. 4.48%). This performance drop is due to the unclear boundary definition. We conclude that our method with uncertainty modeling (class-aware attention) helps to improve the use of narration supervision (action mAP 3.19% v.s. 2.90%). Also, we show that our modeling of class-aware attention is better than the alternative of class-agnostic attention (action mAP 3.19% v.s. 3.09%). The reason, we argue, is that the class-agnostic attention is only able to distinguish the dynamic actions from the background frames (e.g., solving the task in TRUMOS 14 as shown in Tab. 39). It fails if the mixed actions are all semantically meaningful video frames.

Contributions of Multimodal Features. We analyze the contributions of multimodal features via building our models on different subsets of features. We first build our models using single modalities, then present our model considering all types of features.

Table 41: Contributions of multimodal features. We show the Average Precision (%) at certain IoU thresholds (@0.1-@0.5) and the mean Average Precision (Avg.). All numbers are higher the better. The best model is shown in **bold**.

	Action Detection						Verb Detection						Noun Detection					
	@0.1	@0.2	@0.3	@0.4	@0.5	Avg.	@0.1	@0.2	@0.3	@0.4	@0.5	Avg.	@0.1	@0.2	@0.3	@0.4	@0.5	Avg.
RGB	4.49	3.76	2.94	2.25	**1.67**	3.02	8.72	7.07	5.43	4.38	3.15	5.75	**8.70**	**7.13**	**5.29**	**3.71**	**2.56**	**5.48**
Flow	2.32	1.98	1.47	1.10	0.84	1.54	6.59	5.58	4.29	2.95	2.10	4.30	4.33	3.47	2.49	1.68	1.11	2.61
Audio	0.34	0.27	0.23	0.09	0.05	0.20	1.71	1.37	1.07	0.61	0.39	1.03	0.94	0.68	0.51	0.23	0.16	0.50
All	**4.68**	**4.01**	**3.27**	**2.33**	1.65	**3.19**	**9.64**	**7.96**	**6.31**	**4.70**	**3.56**	**6.43**	8.51	6.88	5.09	3.36	2.25	5.22

Tab. 41 shows the results. As for the single modal models, the RGB model provides the best performance on Action Detection (mAP 3.02%). It achieves both high verb detection (mAP 5.75%) and high noun detection (mAP 5.48%) performance. The Flow model (action mAP 1.54%) is worse than the RGB model but better than the Audio model, and we can see that the flow feature provides more information for the dynamic actions (verb mAP 4.30%) while is not that good at localizing static objects temporally (noun mAP 2.61%). The Audio model (action mAP 0.20%) is the worst among the three single modal models, but it still provides useful information, especially in verb detection (mAP 1.03%).

Our final model takes advantage of all features and achieves the best performance in terms of action detection mAP (3.19%). As compared to the RGB model, it utilizes the Flow and Audio information to better detect the dynamic actions (verb mAP 6.43% v.s. 5.75%). Furthermore, as compared to the Flow and Audio models, it combines the appearance feature (RGB) to better recognize objects in the temporal domain (noun mAP 5.22% v.s. 2.61%, 0.50%). In sum, we conclude that our modeling of the multimodal nature of the videos helps to improve the weakly supervised action detection task.

We show in Tab. 42 and Fig. 40 the verb and noun classes most detected by the three modalities. For the verb detection (Fig. 40 (left)), action "wash" can be easily detected by all three modalities, while "fold" only makes a slight sound, so it is hard to be recognized by audio. In comparison, "season" sounds loud, but the dynamic action is nuance; thus, it can be detected by the audio but not the motion flow. The noun detection results is also

Table 42: Top-5 classes detected by the RGB, Flow, and Audio features.

Verb Detection						Noun Detection					
RGB		Flow		Audio		RGB		Flow		Audio	
Name	AP(%)	Name	AP(%)	Name	AP(%)	Name	AP(%)	Name	AP(%)	Name	AP(%)
wash	39.59	wash	37.14	wash	17.47	corn	33.21	yoghurt	28.50	microwave	18.33
filter	31.33	hang	29.69	season	11.81	raisin	33.17	tray	21.81	salt	11.79
rip	30.55	fold	23.32	measure	8.12	yoghurt	33.17	lid	21.14	oatmeal	5.63
season	30.11	dry	19.88	unscrew	6.65	olive	29.34	cloth	20.72	carrot	5.27
fold	25.51	throw	17.75	squeeze	5.11	lid	28.16	oven	18.67	cupboard	4.31

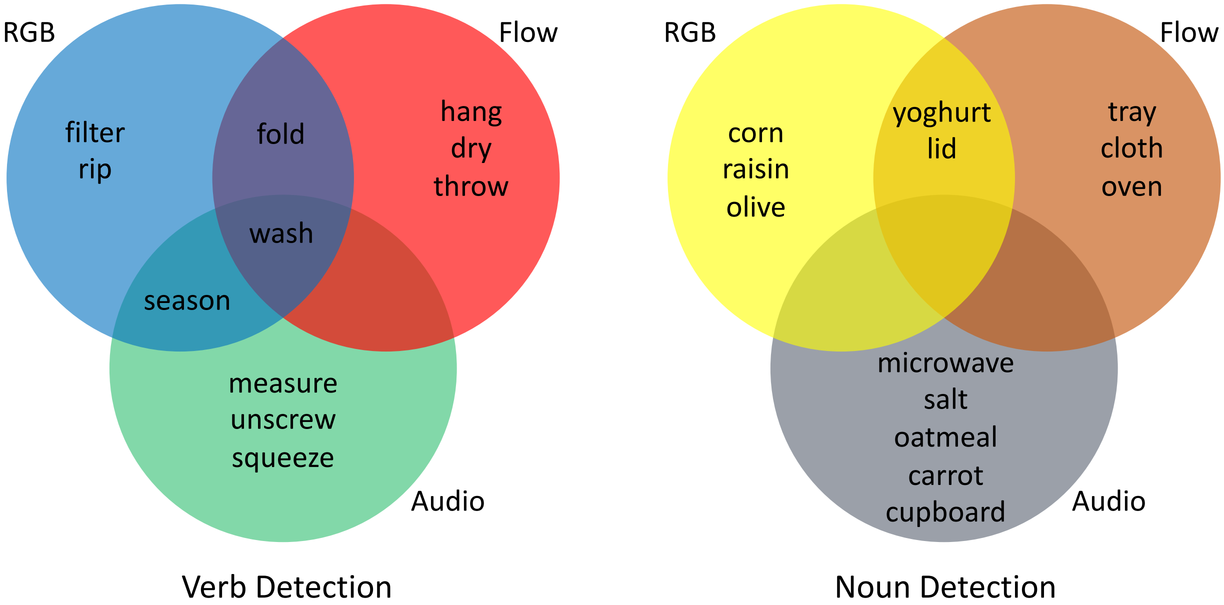


Figure 40: Venn diagrams - The easily detected top-5 classes by different modalities.

interesting (Fig. 40 (right)). We found the “tray” and “cloth” to be more dynamic, and we notice that “microwave” makes a sound. So, we conclude that different modalities help to localize different objects and actions temporally.

9.3.3 Qualitative Examples

We provide a qualitative example visualizing the results of our model. Fig. 41 shows that our model confidently and correctly localizing the actions “wash pan”, “wash spatula”, and

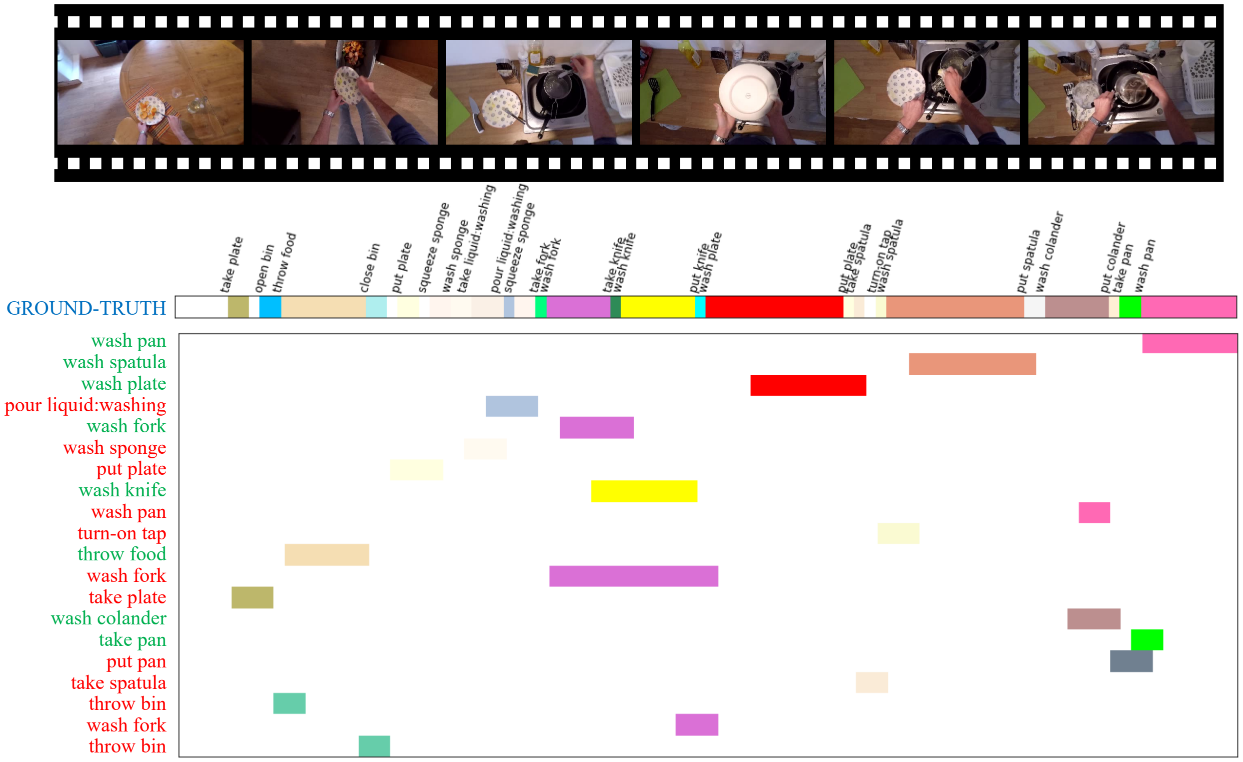


Figure 41: Qualitative example of our model's action detection results. We show the demo of the video, the ground-truth annotations, and our model's top-20 predictions. We show the correct predictions using green and incorrect ones using red. The correctness is determined by IoU@0.5.

"wash plate". For actions such as "pour liquid:washing" and "wash sponge", our model's estimations of the starting and ending time are not precise, thus cause the IoU with the ground truth to be smaller than 0.5. We can hardly find mistakes regarding classification issues in the top-20. Hence we conclude that localization and refining the action boundaries are still challenging for weakly supervised action detection and should be focused on.

9.4 Conclusion

In this chapter, we turn our focus to video action detection in the temporal domain. Compared to the object detection in the images, videos are natural multimodal inputs, and the narration supervision has a more classical characteristic. We developed a model to learn from the narration supervision and utilize multimodal features, including RGB, Motion flow, and ambient sound. In our design, the model learns to attend to the frames related to the narration label while suppressing the irrelevant frames from being used. In the experiments, we show that proposed method outperformed alternative designs. Also, we proved that the different modalities contribute to the detections of different actions and objects, in the temporal domain.

This chapter contributes to the thesis hypothesis H5. It tackles a practical problem of learning action detection model from video sequences using the narration supervision. The audio narrations are imprecise and noisy, but we show our proposed class-aware attention tackles the uncertainty regarding the overlapped actions. We show that our method performs better than directly applying the narration supervision.

Table 43: Conclusion - validated hypotheses in this chapter.

	Multimodal features help to understand images/videos with implicit persuasive intent, such as visual advertisements.	Text features can be unreliable if not modeled appropriately.	Text supervision contains noise, but can be used to localize visual objects in space, if modeled properly.	Text supervision provides contexts regarding visual objects, they are reliable cues for disambiguating entities and relations.	Noisy audio narrations as a multimodal signal can be modeled to localize video actions in temporal domain.
This Chapter					✓

10.0 Conclusions

In this thesis, we explored how intelligent models can perceive the same amount of multimodal knowledge as human beings, and make multimodal inferences based on comprehensive evidence, learn to localize visuals using multimodal supervision. Because the perception gap between human and machine intelligence lies in both the *multimodal inputs* and *multimodal supervision*, we focused on two primary tasks targeting the two — for multimodal inputs, we focused on visual reasoning; for multimodal supervision, we focused on weakly supervised detection.

For both tasks, the first challenge lies in *how to model the multimodal information*. We built basic multimodal models to proceed with visual reasoning with regard to the image/video advertisements (Chapter 3, 4). We show that models can perceive information beyond the traditional modalities such as image and video frames and even rely on knowledge to resolve reasoning tasks. On more general vision tasks, we also trained models to learn from multimodal signals such as text captions (Chapter 7) and audio narrations (Chapter 9). Both works show that multimodal supervision, though weak and noisy, is an efficient way to release the human labor in the annotation-training loop.

Then, we targeted a more practical issue — *noise in the multimodal inputs and supervision*. For visual reasoning (which uses multimodal inputs), we investigated the noise in the external knowledge retrieval process (Chapter 5) and designed a bottom-up hierarchy model to rule out noisy knowledge paragraphs and gradually refine the relevant and useful information. Next, we proposed our observation of the shortcut effects that models over-focus on shallow connections between inputs and outputs in multi-choice VQA. We presented a method to quantify the detrimental effect as well as robust training to alleviate it (Chapter 5, 6). We validated that with the efforts of ruling out noise and shortcut effects, visual reasoning tasks' performance can be improved. For weakly supervised detection (which uses multimodal supervision), we explored the noise in the annotated descriptive captions in Chapter 7. We considered the supervision purity (homogeneity and symmetry scores) of the image-caption pairs in the preprocessing pipeline to filter out or downweigh the noisy examples. Besides, we tackle false negatives (a labeling noise) using a generalizable text classifier in the label extraction process to amplify the visual objects presented in the image but not mentioned in the caption. In Chapter 8, we further explored the way of using caption supervision and proposed to use the text graphs extracted from captions. Such holistic representations help to filter out the noise in the extracted pseudo instance labels (Chapter 7 only associates text entity to arbitrary relevant instance) — a contextualized entity can be more accurately matched to a visual region, providing more reliable cues regarding the location. Finally, we modeled the uncertainty in Chapter 9 to use the noisy audio narration supervision in videos. We assigned class-dependent attention scores to the frames in between the previous and next actions. These scores are proportional to the contribution in the final classification. Hence they can be used to suppress the annotation noise. In sum, we validated in Chapter 7, 8, 9 that explicitly dealing with noise in the supervision also helps to improve the weakly supervised detection models.

10.1 Validated Hypotheses

For the thesis hypotheses, we showed that multimodal features help to understand images/videos with implicit persuasive intent (H1). As demonstrated using the Ads dataset, our model performs better if we consider the slogan/external knowledge/audio/motion (Chapter 3, 4, 5). We then show that special care for the text features is required for reasoning in that they may not always be reliable (H2). We diagnosed the unreliable text features in two datasets and proposed solutions for them: (1) we learned a weighting to de-emphasize unreliable retrieved text knowledge in a graph-alike structure (Chapter 5), and (2) we proposed an approach similar to dropout (Chapter 6) to strengthen vision-language models' robustness. Besides the multimodal inputs, we found that the noisy and unreliable nature of multimodal features also lies in the supervised signals — such as the caption supervision for learning object detection models (H3) and the audio narrations for action detection (H5). Thus, we proposed a weakly supervised action detection model to learn from noisy narrations (Chapter 9) and designed a pipeline to filter, squeeze, amplify, and distill the reliable information from noisy captions (Chapter 7). We further refine reliable evidence from the extra contexts (H4) in captions (Chapter 8) and all these efforts allow us to harvest a robust, accurate, and reliable model.

Table 44: Conclusion - validated hypotheses.

	Multimodal features help to understand images/videos with implicit persuasive intent, such as visual advertisements.	Text features can be unreliable if not modeled appropriately.	Text supervision contains noise, but can be used to localize visual objects in space, if modeled properly.	Text supervision provides contexts regarding visual objects, they are reliable cues for disambiguating entities and relations.	Noisy audio narrations as a multimodal signal can be modeled to localize video actions in temporal domain.
Chapter 3 (ADVISE)	✓		✓		
Chapter 4 (STORY)	✓				✓
Chapter 5 (BREAKING)	✓	✓			
Chapter 6 (VCR)		✓			
Chapter 7 (CAP2DET)			✓		
Chapter 8 (LINGUISTIC)			✓	✓	
Chapter 9 (EPIC)					✓

10.2 Limitations

First, we provide discussion regarding the concerns readers may have, primarily related to the ***low numbers in the experiments*** of Chapter 8, 9. In these two chapters, the prediction is no longer a single classification, but a mixture of categorical and numerical predictions. Given that the final metrics are proportional to the joint probability of making the final correct prediction, and the joint probability are usually much lower than the probability of achieving an individual task (e.g., Chapter 8 requires to correctly predict the subject, predicate, object classes, as well as the subject and object bounding boxes), it is normal that the numbers are low (e.g., R@50, R@100 in Chapter 8, and action detection mAP in Chapter 9). We did not provide the formal significance test to validate the improvements of these low numbers, but the ***experiments are actually reproducible***. We believe that in future work, we can use better metrics that better accord to human intuition and reflect the models' performance.

Then, we have to admit that our methods are still different from human reasoning. In most of our visual reasoning approaches, the neural models are served as black boxes. So, we cannot validate if our models have similar ***decision-making process*** as humans. Though in Chapter 5 we built a hierarchical reasoning graph similar to the human reasoning process to allow the external knowledge to reinforce the prediction, we assumed the hierarchy of the reasoning graph to be known in that we have provided the graph structure (using our human knowledge). This hardcoded graph structure can not adjust itself accordingly even the inputs are changed. So, our models have to rely on a "traditional routine" to reason. Besides, whether human thinks the same way as the routine process needs to be validated.

When we refer to the comparison to human reasoning, the natural issue with regard to ***human performance*** comes. In the thesis, some of our chapters are easy to *compare to the human performance.* For example, in ads understanding, humans can easily catch the points implied in image/video ads, achieving a $> 90\%$ accuracy of choosing the paired statement connected to the ad. In comparison, our best models (chapter 3, 5) only achieved a 87% accuracy for image ads, and only 64% for video ads. The situation of the VCR task is similar (see Chapter 6), humans achieved $> 90\%$ accuracy for both $Q \rightarrow A$ and $QA \rightarrow R$ tasks, while the SOTA model we studied only achieved 68.5%, our robust training improved it by 2% (70.6%). As for the detection models learned from the multimodal supervision (Chapter 7, 8, 9), it is *hard to directly compare their performance to humans.* The primary reason is that detection tasks have an additional variable — location. If we only consider the most confident locations (e.g., top-1, top-2) produced by the models, the result numbers are usually too small to compare since precisely capture the localization information is challenging. In the other end, if we ask humans to provide more than 10 possible predictions to produce metrics such as recall@10, it is a labor-intensive annotation process. To resolve the dilemma, *researchers usually do not require annotators to provide a human upper-bound baseline for detection tasks.* Instead, they treat human annotations as ground-truth, and only compare models' predictions and allow imprecise localization to some extent (e.g., allowing bounding boxes > 50% overlapped to the ground-truth to be treated as correct). Though this evaluation did not measure the performance gap between humans and machines, it can be *used for comparing different models*. We ensured in each chapter that our ***comparison to the baselines is fair***.

Next, for the random noise we studied (Chapter 5, 7), a direction that is less explored is the ***modeling of the probability distribution***. For example, whether the caption describes the visual objects (Chapter 7) may be conditional depending on the image-caption relevance, the category of the visual objects, and even the size of the objects. Therefore, it is possible to design a more dedicated noise model considering all these factors beyond the homogeneity and symmetry scores. We can even design a mechanism to filter out the noisy image-caption pairs (Chapter 7) or exclude some knowledge connections (Chapter 5) based on the modeled probability distributions during training.

We provide some brainstorming regarding the ***more preferable systematic solutions*** of using text captions or audio narrations to train detection models. In our Chapter 7, 8, 9, we proposed pipelines for learning image/video detection models from text captions or audio narrations. However, these pipelines only simply connect the different processing steps, without a holistic consideration. For example: (1) Chapter 7 requires to separately train a model to produce the homogeneity and symmetry scores, and a text classifier to predict the presence of visual objects given the caption; (2) in Chapter 8, the text graph is extracted by an external model; (3) the visual region proposals used in Chapter 7, 8 are from pre-trained region proposal networks. These separately trained components bring helpful domain knowledge but also the confusing ones since they are optimized on the original tasks but not the vision-related ones. For example, we have seen noisy and imprecise subject-predicate-object triplets such as "group-of-people". Thus, we imagine a unified system that can take advantage of the external module while also feeding back to these components, telling them to refresh their knowledge gathered on the original domain. For example, "group-of-people" should be adjusted to a single entity "people", while the region proposals can be refined by their relevance to the caption.

Finally, we would envision the possibility of ***using the linguistic structures of the narrations*** in the EPIC Kitchens dataset. We based our Chapter 9 on the verb+noun parsed from the language parser (see [45]). Hence, the chapter is analogous to Chapter 7 which extracts object categories from descriptive captions. However, it is also possible to parse the closed captions into text graphs to represent the video contents (similar to Chapter 8). In this way, we can model the temporal dependency of the different objects/actions and utilize temporal context to better localizing them.

10.3 Broader Impacts

In sum, we have made progress to allow machine intelligence to perceive multimodal information. Though there are limitations, we are promising about the broader impacts of the thesis study. We assume the AI agent in the multimodal environment will be the perfect testbed for the technology, such as the self-driving vehicles. Using the multimodal inputs, vehicles can drive in different weather and daylight conditions because the lidar and radar sensors complement the visual signals; the driving commands can also be entered by speech since the machine intelligence can parse the commands and incorporates both knowledge and environmental information to make decisions. Moreover, using the multimodal signals for training further alleviates the efforts of humans in the annotation-training loop, allowing faster iteration of the new products or new applications, or at least for testing new ideas. In the long future, smart vehicles may even learn from the raw signals, recognizing the concepts of traffic signs from pixels, building precise estimation regarding the scene depths, and learning the driving behavior — while all without concrete supervision.

Bibliography

[1] Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, et al. Tensorflow: A system for large-scale machine learning. In *OSDI*, volume 16, pages 265–283, 2016.

[2] Triantafyllos Afouras, Andrew Owens, Joon Son Chung, and Andrew Zisserman. Self-supervised learning of audio-visual objects from video. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XVIII 16*, pages 208–224. Springer, 2020.

[3] Aishwarya Agrawal, Dhruv Batra, Devi Parikh, and Aniruddha Kembhavi. Don't just assume; look and answer: Overcoming priors for visual question answering. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[4] Karuna Ahuja, Karan Sikka, Anirban Roy, and Ajay Divakaran. Understanding visual ads by aligning symbols and objects using co-attention. In *CVPR Workshop towards Automatic Understanding of Visual Advertisements*, 2018.

[5] Chris Alberti, Jeffrey Ling, Michael Collins, and David Reitter. Fusion of detected objects in text for visual question answering. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2131–2140, Hong Kong, China, November 2019. Association for Computational Linguistics.

[6] Brandon Amos, Bartosz Ludwiczuk, and Mahadev Satyanarayanan. Openface: A general-purpose face recognition library with mobile applications. *CMU School of Computer Science*, 2016.

[7] 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 *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[8] Jacob Andreas, Marcus Rohrbach, Trevor Darrell, and Dan Klein. Learning to compose neural networks for question answering. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1545–1554, San Diego, California, June 2016. Association for Computational Linguistics.

[9] Jacob Andreas, Marcus Rohrbach, Trevor Darrell, and Dan Klein. Neural module networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[10] Gabor Angeli, Melvin Jose Johnson Premkumar, and Christopher D. Manning. Leveraging linguistic structure for open domain information extraction. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, July 2015.

[11] 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 *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[12] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C. Lawrence Zitnick, and Devi Parikh. Vqa: Visual question answering. In *The IEEE International Conference on Computer Vision (ICCV)*, 2015.

[13] Relja Arandjelovic and Andrew Zisserman. Look, listen and learn. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[14] Relja Arandjelovic and Andrew Zisserman. Objects that sound. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018.

[15] Yusuf Aytar, Carl Vondrick, and Antonio Torralba. Soundnet: Learning sound representations from unlabeled video. In *Advances in Neural Information Processing Systems*, 2016.

[16] Javad Azimi, Ruofei Zhang, Yang Zhou, Vidhya Navalpakkam, Jianchang Mao, and Xiaoli Fern. Visual appearance of display ads and its effect on click through rate. In *Proceedings of the ACM International Conference on Information and Knowledge Management*, 2012.

[17] Peter W Battaglia, Jessica B Hamrick, Victor Bapst, Alvaro Sanchez-Gonzalez, Vinicius Zambaldi, Mateusz Malinowski, Andrea Tacchetti, David Raposo, Adam Santoro, Ryan Faulkner, et al. Relational inductive biases, deep learning, and graph networks. *arXiv preprint arXiv:1806.01261*, 2018.

[18] Judy Battista. Roger goodell, nfl rightly correct course with change in policy, August 2014. `http://www.nfl.com/news/story/0ap3000000384987/article/roger-goodell-nfl-rightly-correct-course-with-change-in-policy`.

[19] Amy Bearman, Olga Russakovsky, Vittorio Ferrari, and Li Fei-Fei. What's the point: Semantic segmentation with point supervision. In *European conference on computer vision*, pages 549–565. Springer, 2016.

[20] Carlos Fabian Benitez-Quiroz, Ramprakash Srinivasan, Aleix M Martinez, et al. Emotionet: An accurate, real-time algorithm for the automatic annotation of a million facial expressions in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5562–5570, 2016.

[21] Alexander C Berg, Tamara L Berg, Hal Daume, Jesse Dodge, Amit Goyal, Xufeng Han, Alyssa Mensch, Margaret Mitchell, Aneesh Sood, Karl Stratos, et al. Understanding and predicting importance in images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2012.

[22] Hakan Bilen and Andrea Vedaldi. Weakly supervised deep detection networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[23] Gary Bradski. The opencv library. *Dr. Dobb's Journal of Software Tools*, 2000.

[24] Thomas Brox, Andrés Bruhn, Nils Papenberg, and Joachim Weickert. High accuracy optical flow estimation based on a theory for warping. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 25–36. Springer, 2004.

[25] Zoya Bylinskii, Sami Alsheikh, Spandan Madan, Adria Recasens, Kimberli Zhong, Hanspeter Pfister, Fredo Durand, and Aude Oliva. Understanding infographics through textual and visual tag prediction. *arXiv preprint arXiv:1709.09215*, 2017.

[26] Fabian Caba Heilbron, Victor Escorcia, Bernard Ghanem, and Juan Carlos Niebles. Activitynet: A large-scale video benchmark for human activity understanding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

[27] Yue Cao, Mingsheng Long, Jianmin Wang, and Shichen Liu. Deep visual-semantic quantization for efficient image retrieval. In *CVPR*, 2017.

[28] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4724–4733. IEEE, 2017.

[29] Brandon Castellano. Pyscenedetect. `https://github.com/Breakthrough/PySceneDetect/`.

[30] Lluis Castrejon, Yusuf Aytar, Carl Vondrick, Hamed Pirsiavash, and Antonio Torralba. Learning aligned cross-modal representations from weakly aligned data. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[31] Kan Chen, Trung Bui, Chen Fang, Zhaowen Wang, and Ram Nevatia. Amc: Attention guided multi-modal correlation learning for image search. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[32] Kan Chen, Jiyang Gao, and Ram Nevatia. Knowledge aided consistency for weakly supervised phrase grounding. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[33] Liang-Chieh Chen, Yi Yang, Jiang Wang, Wei Xu, and Alan L. Yuille. Attention to scale: Scale-aware semantic image segmentation. In *Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2016.

[34] Tianshui Chen, Weihao Yu, Riquan Chen, and Liang Lin. Knowledge-embedded routing network for scene graph generation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[35] Tseng-Hung Chen, Yuan-Hong Liao, Ching-Yao Chuang, Wan-Ting Hsu, Jianlong Fu, and Min Sun. Show, adapt and tell: Adversarial training of cross-domain image captioner. In *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[36] 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. *arXiv preprint arXiv:1504.00325*, 2015.

[37] Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. Uniter: Learning universal image-text representations. *arXiv preprint arXiv:1909.11740*, 2019.

[38] Yuhua Chen, Wen Li, Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Domain adaptive faster r-cnn for object detection in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[39] Haibin Cheng, Roelof van Zwol, Javad Azimi, Eren Manavoglu, Ruofei Zhang, Yang Zhou, and Vidhya Navalpakkam. Multimedia features for click prediction of new ads in display advertising. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2012.

[40] Jinsoo Choi, Tae-Hyun Oh, and In So Kweon. Video-story composition via plot analysis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[41] Kevin Clark, Minh-Thang Luong, Quoc V Le, and Christopher D Manning. Electra: Pre-training text encoders as discriminators rather than generators. In *International Conference on Learning Representations*, 2020.

[42] Ira Cohen, Nicu Sebe, Ashutosh Garg, Lawrence S Chen, and Thomas S Huang. Facial expression recognition from video sequences: temporal and static modeling. *Computer Vision and image understanding*, 91(1-2):160–187, 2003.

[43] Ben Cosgrove. The photo that changed the face of aids, November 2014. `http://time.com/3503000/behind-the-picture-the-photo-that-changed-the-face-of-aids/`.

[44] Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Antonino Furnari, Evangelos Kazakos, Jian Ma, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, et al. Rescaling egocentric vision. *arXiv preprint arXiv:2006.13256*, 2020.

[45] Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Sanja Fidler, Antonino Furnari, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, and Michael Wray. Scaling egocentric vision: The epic-kitchens dataset. In *The European Conference on Computer Vision (ECCV)*, September 2018.

[46] Luciano Del Corro and Rainer Gemulla. Clausie: clause-based open information extraction. In *Proceedings of the International Conference on World Wide Web (WWW)*, 2013.

[47] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2009.

[48] Karan Desai and Justin Johnson. VirTex: Learning Visual Representations from Textual Annotations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2021.

[49] FFmpeg developers. Ffmpeg. `https://www.ffmpeg.org/`.

[50] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, June 2019.

[51] Ali Diba, Vivek Sharma, Ali Pazandeh, Hamed Pirsiavash, and Luc Van Gool. Weakly supervised cascaded convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[52] 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 *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

[53] Aviv Eisenschtat and Lior Wolf. Linking image and text with 2-way nets. In *CVPR*, 2017.

[54] Irfan A. Essa and Alex Paul Pentland. Coding, analysis, interpretation, and recognition of facial expressions. *IEEE transactions on pattern analysis and machine intelligence*, 19(7):757–763, 1997.

[55] Anthony Fader, Stephen Soderland, and Oren Etzioni. Identifying relations for open information extraction. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, July 2011.

[56] Fartash Faghri, David J Fleet, Jamie Ryan Kiros, and Sanja Fidler. Vse++: Improved visual-semantic embeddings. *arXiv preprint arXiv:1707.05612*, 2017.

[57] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.

[58] David Fleet and Yair Weiss. Optical flow estimation. In *Handbook of mathematical models in computer vision*, pages 237–257. Springer, 2006.

[59] Forbes. What makes a tv commercial memorable and effective?, 2012.

[60] Gustav Freytag. *Freytag's technique of the drama: an exposition of dramatic composition and art.* Scholarly Press, 1896.

[61] Jianlong Fu, Heliang Zheng, and Tao Mei. Look closer to see better: recurrent attention convolutional neural network for fine-grained image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[62] Akira Fukui, Dong Huk Park, Daylen Yang, Anna Rohrbach, Trevor Darrell, and Marcus Rohrbach. Multimodal compact bilinear pooling for visual question answering and visual grounding. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 457–468, Austin, Texas, November 2016. Association for Computational Linguistics.

[63] Antonino Furnari and Giovanni Maria Farinella. What would you expect? anticipating egocentric actions with rolling-unrolling lstms and modality attention. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.

[64] Chuang Gan, Zhe Gan, Xiaodong He, Jianfeng Gao, and Li Deng. Stylenet: Generating attractive visual captions with styles. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[65] Zhe Gan, Yen-Chun Chen, Linjie Li, Chen Zhu, Yu Cheng, and Jingjing Liu. Large-scale adversarial training for vision-and-language representation learning. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2020.

[66] Ruohan Gao, Rogerio Feris, and Kristen Grauman. Learning to separate object sounds by watching unlabeled video. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018.

[67] John M Gauch and Abhishek Shivadas. Finding and identifying unknown commercials using repeated video sequence detection. *Computer Vision and Image Understanding*, 103(1):80–88, 2006.

[68] Jort F Gemmeke, Daniel PW Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R Channing Moore, Manoj Plakal, and Marvin Ritter. Audio set: An ontology and human-labeled dataset for audio events. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 776–780. IEEE, 2017.

[69] Ross Girshick. Fast r-cnn. In *The IEEE International Conference on Computer Vision (ICCV)*, 2015.

[70] 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*, pages 580–587, 2014.

[71] Georgia Gkioxari and Jitendra Malik. Finding action tubes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

[72] Omer Goldman, Veronica Latcinnik, Ehud Nave, Amir Globerson, and Jonathan Berant. Weakly supervised semantic parsing with abstract examples. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, July 2018.

[73] Wonjoon Goo, Juyong Kim, Gunhee Kim, and Sung Ju Hwang. Taxonomy-regularized semantic deep convolutional neural networks. In *European Conference on Computer Vision*, pages 86–101. Springer, 2016.

[74] Google. Google cloud speech-to-text api. `https://cloud.google.com/speech-to-text/`.

[75] Google. Google cloud vision api. `https://cloud.google.com/vision/`.

[76] Raghuraman Gopalan, Ruonan Li, and Rama Chellappa. Domain adaptation for object recognition: An unsupervised approach. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, November 2011.

[77] 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 *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[78] Jiuxiang Gu, Handong Zhao, Zhe Lin, Sheng Li, Jianfei Cai, and Mingyang Ling. Scene graph generation with external knowledge and image reconstruction. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[79] Danna Gurari, Qing Li, Abigale J Stangl, Anhong Guo, Chi Lin, Kristen Grauman, Jiebo Luo, and Jeffrey P Bigham. Vizwiz grand challenge: Answering visual questions from blind people. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3608–3617, 2018.

[80] David Harwath, Adria Recasens, Didac Suris, Galen Chuang, Antonio Torralba, and James Glass. Jointly discovering visual objects and spoken words from raw sensory input. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018.

[81] Kaiming He, Georgia Gkioxari, Piotr Dollar, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, October 2017.

[82] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[83] Lisa Anne Hendricks, Zeynep Akata, Marcus Rohrbach, Jeff Donahue, Bernt Schiele, and Trevor Darrell. Generating visual explanations. In *European Conference on Computer Vision*, 2016.

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

[85] Nigel Hollis. Why good advertising works (even when you think it doesn't), August 2011. `https://www.theatlantic.com/business/archive/2011/08/why-good-advertising-works-even-when-you-think-it-doesnt/244252/`.

[86] Murhaf Hossari, Soumyabrata Dev, Matthew Nicholson, Killian McCabe, Atul Nautiyal, Clare Conran, Jian Tang, Wei Xu, and François Pitié. Adnet: A deep network for detecting adverts. In *26th AIAI Irish Conference on Artificial Intelligence and Cognitive Science (AICS)*, 2018.

[87] Harold Hotelling. Relations between two sets of variates. *Biometrika*, 28(3/4):321–377, 1936.

[88] Ronghang Hu, Jacob Andreas, Marcus Rohrbach, Trevor Darrell, and Kate Saenko. Learning to reason: End-to-end module networks for visual question answering. In *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[89] Jonathan Huang, Vivek Rathod, Chen Sun, Menglong Zhu, Anoop Korattikara, Alireza Fathi, Ian Fischer, Zbigniew Wojna, Yang Song, Sergio Guadarrama, and Kevin Murphy. Speed/accuracy trade-offs for modern convolutional object detectors. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[90] Yao-Hung Hubert Tsai, Liang-Kang Huang, and Ruslan Salakhutdinov. Learning robust visual-semantic embeddings. In *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[91] Drew A. Hudson and Christopher D. Manning. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[92] Mark J Huiskes, Bart Thomee, and Michael S Lew. New trends and ideas in visual concept detection: the mir flickr retrieval evaluation initiative. In *Proceedings of the international conference on Multimedia information retrieval*, pages 527–536. ACM, 2010.

[93] 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 *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[94] Zaeem Hussain, Mingda Zhang, Xiaozhong Zhang, Keren Ye, Christopher Thomas, Zuha Agha, Nathan Ong, and Adriana Kovashka. Automatic understanding of image and video advertisements. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[95] Haroon Idrees, Amir R Zamir, Yu-Gang Jiang, Alex Gorban, Ivan Laptev, Rahul Sukthankar, and Mubarak Shah. The thumos challenge on action recognition for videos “in the wild”. *Computer Vision and Image Understanding*, 155:1–23, 2017.

[96] Eddy Ilg, Nikolaus Mayer, Tonmoy Saikia, Margret Keuper, Alexey Dosovitskiy, and Thomas Brox. Flownet 2.0: Evolution of optical flow estimation with deep networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 2, 2017.

[97] Mohit Iyyer, Varun Manjunatha, Anupam Guha, Yogarshi Vyas, Jordan Boyd-Graber, Hal Daume, III, and Larry S. Davis. The amazing mysteries of the gutter: Drawing inferences between panels in comic book narratives. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[98] Max Jaderberg, Karen Simonyan, Andrew Zisserman, et al. Spatial transformer networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2015.

[99] Mihir Jain, Jan van Gemert, Herve Jegou, Patrick Bouthemy, and Cees G.M. Snoek. Action localization with tubelets from motion. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2014.

[100] Achiya Jerbi, Roei Herzig, Jonathan Berant, Gal Chechik, and Amir Globerson. Learning object detection from captions via textual scene attributes. *arXiv preprint arXiv:2009.14558*, 2020.

[101] Robin Jia and Percy Liang. Adversarial examples for evaluating reading comprehension systems. In *The 2017 Conference on Empirical Methods in Natural Language Processing*, 2017.

[102] Lu Jiang, Deyu Meng, Qian Zhao, Shiguang Shan, and Alexander G Hauptmann. Self-paced curriculum learning. In *Twenty-Ninth Association for the Advancement of Artificial Intelligence (AAAI) Conference on Artificial Intelligence*, volume 2, page 6, 2015.

[103] Lu Jiang, Zhengyuan Zhou, Thomas Leung, Li-Jia Li, and Li Fei-Fei. Mentornet: Learning data-driven curriculum for very deep neural networks on corrupted labels. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 2309–2318, 2018.

[104] Yichen Jiang and Mohit Bansal. Avoiding reasoning shortcuts: Adversarial evaluation, training, and model development for multi-hop QA. In *The 57th Annual Meeting of the Association for Computational Linguistics*, 2019.

[105] 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 *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[106] 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 (ICCV)*, Oct 2017.

[107] Justin Johnson, Andrej Karpathy, and Li Fei-Fei. Densecap: Fully convolutional localization networks for dense captioning. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[108] Justin Johnson, Ranjay Krishna, Michael Stark, Li-Jia Li, David Shamma, Michael Bernstein, and Li Fei-Fei. Image retrieval using scene graphs. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

[109] Jungseock Joo, Weixin Li, Francis F Steen, and Song-Chun Zhu. Visual persuasion: Inferring communicative intents of images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[110] Jungseock Joo, Francis F Steen, and Song-Chun Zhu. Automated facial trait judgment and election outcome prediction: Social dimensions of face. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3712–3720, 2015.

[111] Takeo Kanade, Jeffrey F Cohn, and Yingli Tian. Comprehensive database for facial expression analysis. In *Automatic Face and Gesture Recognition, 2000. Proceedings. Fourth IEEE International Conference on*, pages 46–53. IEEE, 2000.

[112] Kai Kang, Hongsheng Li, Junjie Yan, Xingyu Zeng, Bin Yang, Tong Xiao, Cong Zhang, Zhe Wang, Ruohui Wang, Xiaogang Wang, et al. T-cnn: Tubelets with convolutional neural networks for object detection from videos. *IEEE Transactions on Circuits and Systems for Video Technology*, 28(10):2896–2907, 2017.

[113] Kai Kang, Wanli Ouyang, Hongsheng Li, and Xiaogang Wang. Object detection from video tubelets with convolutional neural networks. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[114] Vadim Kantorov, Maxime Oquab, Minsu Cho, and Ivan Laptev. Contextlocnet: Context-aware deep network models for weakly supervised localization. In *Proceedings of the European Conference on Computer Vision (ECCV)*, October 2016.

[115] Andrej Karpathy and Li Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

[116] Andrej Karpathy, Armand Joulin, and Li F Fei-Fei. Deep fragment embeddings for bidirectional image sentence mapping. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2014.

[117] Aniruddha Kembhavi, Mike Salvato, Eric Kolve, Minjoon Seo, Hannaneh Hajishirzi, and Ali Farhadi. A diagram is worth a dozen images. In *European Conference on Computer Vision*, pages 235–251. Springer, 2016.

[118] Eun-Sol Kim, Woo Young Kang, Kyoung-Woon On, Yu-Jung Heo, and Byoung-Tak Zhang. Hypergraph attention networks for multimodal learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[119] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *International Conference for Learning Representations (ICLR)*, 2015.

[120] Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *5th International Conference on Learning Representations (ICLR)*, 2017.

[121] Ryan Kiros, Ruslan Salakhutdinov, and Richard S Zemel. Unifying visual-semantic embeddings with multimodal neural language models. In *TACL*, 2015.

[122] Ronak Kosti, Jose M Alvarez, Adria Recasens, and Agata Lapedriza. Emotic: Emotions in context dataset. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2017 IEEE Conference on*, pages 2309–2317. IEEE, 2017.

[123] Adriana Kovashka and James Hahn. Automatic understanding of visual advertisements. `https://evalai.cloudcv.org/web/challenges/challenge-page/86/overview`.

[124] Jonathan Krause, Justin Johnson, Ranjay Krishna, and Li Fei-Fei. A hierarchical approach for generating descriptive image paragraphs. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[125] Ranjay Krishna, Kenji Hata, Frederic Ren, Li Fei-Fei, and Juan Carlos Niebles. Dense-captioning events in videos. In *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[126] 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. *International Journal of Computer Vision*, 123(1):32–73, 2017.

[127] S. Shunmuga Krishnan and Ramesh K. Sitaraman. Understanding the effectiveness of video ads: A measurement study. In *Proceedings of the 2013 Conference on Internet Measurement Conference*, IMC '13, pages 149–162, New York, NY, USA, 2013. ACM.

[128] Julia Kruk, Jonah Lubin, Karan Sikka, Xiao Lin, Dan Jurafsky, and Ajay Divakaran. Integrating text and image: Determining multimodal document intent in instagram posts. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, 2019.

[129] Krishna Kumar Singh and Yong Jae Lee. Hide-and-seek: Forcing a network to be meticulous for weakly-supervised object and action localization. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[130] Krishna Kumar Singh, Fanyi Xiao, and Yong Jae Lee. Track and transfer: Watching videos to simulate strong human supervision for weakly-supervised object detection. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[131] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, et al. The open images dataset v4. *International Journal of Computer Vision (IJCV)*, 2020.

[132] Suha Kwak, Minsu Cho, Ivan Laptev, Jean Ponce, and Cordelia Schmid. Unsupervised object discovery and tracking in video collections. In *The IEEE International Conference on Computer Vision (ICCV)*, December 2015.

[133] Jun-Tae Lee, Mihir Jain, Hyoungwoo Park, and Sungrack Yun. Cross-attentional audio-visual fusion for weakly-supervised action localization. In *International Conference on Learning Representations*, 2020.

[134] Jens Lehmann, Robert Isele, Max Jakob, Anja Jentzsch, Dimitris Kontokostas, Pablo N Mendes, Sebastian Hellmann, Mohamed Morsey, Patrick Van Kleef, Sören Auer, et al. Dbpedia–a large-scale, multilingual knowledge base extracted from wikipedia. *Semantic Web*, 2015.

[135] James H Leigh and Terrance G Gabel. Symbolic interactionism: its effects on consumer behaviour and implications for marketing strategy. *Journal of Services Marketing*, 6(3):5–16, 1992.

[136] Sidney J Levy. Symbols for sale. *Harvard business review*, 37(4):117–124, 1959.

[137] Dong Li, Jia-Bin Huang, Yali Li, Shengjin Wang, and Ming-Hsuan Yang. Weakly supervised object localization with progressive domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[138] Gen Li, Nan Duan, Yuejian Fang, Daxin Jiang, and Ming Zhou. Unicoder-vl: A universal encoder for vision and language by cross-modal pre-training. *arXiv preprint arXiv:1908.06066*, 2019.

[139] Hao Li and Hui-Yi Lo. Do you recognize its brand? the effectiveness of online in-stream video advertisements. *Journal of Advertising*, 44(3):208–218, 2015.

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

[141] Xuelong Li, Di Hu, and Xiaoqiang Lu. Image2song: Song retrieval via bridging image content and lyric words. In *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[142] Zhe Li, Yazan Abu Farha, and Jurgen Gall. Temporal action segmentation from timestamp supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8365–8374, June 2021.

[143] Karin Liebhart and Petra Bernhardt. Political storytelling on instagram: Key aspects of alexander van der bellen's successful 2016 presidential election campaign. *Media and Communication*, 5(4):15–25, 2017.

[144] Jingxiang Lin, Unnat Jain, and Alexander Schwing. Tab-vcr: Tags and attributes based vcr baselines. In *Advances in Neural Information Processing Systems 32*, pages 15615–15628. Curran Associates, Inc., 2019.

[145] Tianwei Lin, Xiao Liu, Xin Li, Errui Ding, and Shilei Wen. Bmn: Boundary-matching network for temporal action proposal generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.

[146] Tianwei Lin, Xu Zhao, Haisheng Su, Chongjing Wang, and Ming Yang. Bsn: Boundary sensitive network for temporal action proposal generation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018.

[147] 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*, pages 740–755. Springer, 2014.

[148] Xin Lin, Changxing Ding, Jinquan Zeng, and Dacheng Tao. Gps-net: Graph property sensing network for scene graph generation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[149] Anan Liu, Jintao Li, Yongdong Zhang, Sheng Tang, Yan Song, and Zhaoxuan Yang. An innovative model of tempo and its application in action scene detection for movie analysis. In *WACV*, 2008.

[150] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *European conference on computer vision*, pages 21–37. Springer, 2016.

[151] Cewu Lu, Ranjay Krishna, Michael Bernstein, and Li Fei-Fei. Visual relationship detection with language priors. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2016.

[152] Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. In *Advances in Neural Information Processing Systems*, 2019.

[153] Jiasen Lu, Caiming Xiong, Devi Parikh, and Richard Socher. Knowing when to look: Adaptive attention via a visual sentinel for image captioning. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[154] 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 (NIPS)*, 2016.

[155] Jiasen Lu, Jianwei Yang, Dhruv Batra, and Devi Parikh. Neural baby talk. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[156] Jiaying Lu, Xin Ye, Yi Ren, and Yezhou Yang. Good, better, best: Textual distractors generation for multi-choice vqa via policy gradient. *arXiv preprint arXiv:1910.09134*, 2019.

[157] Shugao Ma, Leonid Sigal, and Stan Sclaroff. Learning activity progression in lstms for activity detection and early detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[158] Xiaofei Ma, Peng Xu, Zhiguo Wang, Ramesh Nallapati, and Bing Xiang. Domain adaptation with bert-based domain classification and data selection. In *Proceedings of the 2nd Workshop on Deep Learning Approaches for Low-Resource NLP (DeepLo 2019)*, pages 76–83, 2019.

[159] Vijay Mahadevan, Weixin Li, Viral Bhalodia, and Nuno Vasconcelos. Anomaly detection in crowded scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1975–1981. IEEE, 2010.

[160] Long Mai, Hailin Jin, Zhe Lin, Chen Fang, Jonathan Brandt, and Feng Liu. Spatial-semantic image search by visual feature synthesis. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[161] Mateusz Malinowski, Marcus Rohrbach, and Mario Fritz. Ask your neurons: A neural-based approach to answering questions about images. In *The IEEE International Conference on Computer Vision (ICCV)*, December 2015.

[162] Jiayuan Mao, Chuang Gan, Pushmeet Kohli, Joshua B. Tenenbaum, and Jiajun Wu. The Neuro-Symbolic Concept Learner: Interpreting Scenes, Words, and Sentences

From Natural Supervision. In *International Conference on Learning Representations*, 2019.

[163] Kenneth Marino, Mohammad Rastegari, Ali Farhadi, and Roozbeh Mottaghi. Okvqa: A visual question answering benchmark requiring external knowledge. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[164] Kenneth Marino, Ruslan Salakhutdinov, and Abhinav Gupta. The more you know: Using knowledge graphs for image classification. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[165] Mausam, Michael Schmitz, Stephen Soderland, Robert Bart, and Oren Etzioni. Open language learning for information extraction. In *Proceedings of the Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, July 2012.

[166] Nikolaus Mayer, Eddy Ilg, Philip Hausser, Philipp Fischer, Daniel Cremers, Alexey Dosovitskiy, and Thomas Brox. A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4040–4048, 2016.

[167] Daniel McDuff, Rana El Kaliouby, Jeffrey F Cohn, and Rosalind Picard. Predicting ad liking and purchase intent: Large-scale analysis of facial responses to ads. *IEEE Transactions on Affective Computing*, 2014.

[168] Tao Mei, Lusong Li, Xian-Sheng Hua, and Shipeng Li. Imagesense: towards contextual image advertising. *ACM Transactions on Multimedia Computing, Communications, and Applications*, 8(1):6, 2012.

[169] Merriam-webster.com. `https://www.merriam-webster.com/dictionary/climax`.

[170] Paul Messaris. *Visual persuasion: The role of images in advertising.* Sage, 1997.

[171] Antoine Miech, Jean-Baptiste Alayrac, Lucas Smaira, Ivan Laptev, Josef Sivic, and Andrew Zisserman. End-to-end learning of visual representations from uncurated instructional videos. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[172] Antoine Miech, Dimitri Zhukov, Jean-Baptiste Alayrac, Makarand Tapaswi, Ivan Laptev, and Josef Sivic. Howto100m: Learning a text-video embedding by watch-

ing hundred million narrated video clips. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, October 2019.

[173] 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.

[174] Ishan Misra, Abhinav Gupta, and Martial Hebert. From red wine to red tomato: Composition with context. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[175] 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)*, June 2016.

[176] Ishan Misra, Abhinav Shrivastava, and Martial Hebert. Watch and learn: Semi-supervised learning for object detectors from video. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

[177] Niluthpol Chowdhury Mithun, Sujoy Paul, and Amit K. Roy-Chowdhury. Weakly supervised video moment retrieval from text queries. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[178] Ali Mollahosseini, Behzad Hasani, and Mohammad H Mahoor. Affectnet: A database for facial expression, valence, and arousal computing in the wild. *IEEE Transactions on Affective Computing*, 2017.

[179] Davide Moltisanti, Sanja Fidler, and Dima Damen. Action recognition from single timestamp supervision in untrimmed videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[180] Yair Movshovitz-Attias, Alexander Toshev, Thomas K Leung, Sergey Ioffe, and Saurabh Singh. No fuss distance metric learning using proxies. In *ICCV*, 2017.

[181] Caroline Lego Muñoz and Terri L Towner. The image is the message: Instagram marketing and the 2016 presidential primary season. *Journal of Political Marketing*, 16(3-4):290–318, 2017.

[182] Seil Na, Sangho Lee, Jisung Kim, and Gunhee Kim. A read-write memory network for movie story understanding. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[183] Arsha Nagrani, Chen Sun, David Ross, Rahul Sukthankar, Cordelia Schmid, and Andrew Zisserman. Speech2action: Cross-modal supervision for action recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[184] Hyeonseob Nam, Jung-Woo Ha, and Jeonghee Kim. Dual attention networks for multimodal reasoning and matching. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[185] Medhini Narasimhan, Svetlana Lazebnik, and Alexander Schwing. Out of the box: Reasoning with graph convolution nets for factual visual question answering. In *Advances in Neural Information Processing Systems*, 2018.

[186] Medhini Narasimhan and Alexander G. Schwing. Straight to the facts: Learning knowledge base retrieval for factual visual question answering. In *The European Conference on Computer Vision (ECCV)*, September 2018.

[187] Alejandro Newell and Jia Deng. Pixels to graphs by associative embedding. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

[188] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked hourglass networks for human pose estimation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 483–499. Springer, 2016.

[189] Jiquan Ngiam, Aditya Khosla, Mingyu Kim, Juhan Nam, Honglak Lee, and Andrew Y Ng. Multimodal deep learning. In *ICML*, 2011.

[190] Phuc Nguyen, Ting Liu, Gautam Prasad, and Bohyung Han. Weakly supervised action localization by sparse temporal pooling network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[191] Phuc Xuan Nguyen, Deva Ramanan, and Charless C. Fowlkes. Weakly-supervised action localization with background modeling. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.

[192] Zhenxing Niu, Mo Zhou, Le Wang, Xinbo Gao, and Gang Hua. Hierarchical multimodal lstm for dense visual-semantic embedding. In *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[193] Megan O'Neill. Old spice response campaign was more popular than obama, August 2015. `https://www.adweek.com/digital/old-spice-response-campaign/`.

[194] Maxime Oquab, Leon Bottou, Ivan Laptev, and Josef Sivic. Is object localization for free? - weakly-supervised learning with convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

[195] Mayu Otani, Yuki Iwazaki, and Kota Yamaguchi. Unreasonable effectiveness of ocr in visual advertisement understanding. In *CVPR Workshop towards Automatic Understanding of Visual Advertisements*, 2018.

[196] Andrew Owens, Phillip Isola, Josh McDermott, Antonio Torralba, Edward H. Adelson, and William T. Freeman. Visually indicated sounds. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[197] Sujoy Paul, Sourya Roy, and Amit K. Roy-Chowdhury. W-talc: Weakly-supervised temporal activity localization and classification. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018.

[198] Marco Pedersoli, Thomas Lucas, Cordelia Schmid, and Jakob Verbeek. Areas of attention for image captioning. In *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[199] Yilang Peng. Same candidates, different faces: Uncovering media bias in visual portrayals of presidential candidates with computer vision. *Journal of Communication*, 68(5):920–941, 2018.

[200] Jeffrey Pennington, Richard Socher, and Christopher Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014.

[201] Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long*

Papers), pages 2227–2237, New Orleans, Louisiana, June 2018. Association for Computational Linguistics.

[202] Julia Peyre, Josef Sivic, Ivan Laptev, and Cordelia Schmid. Weakly-supervised learning of visual relations. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[203] Mengshi Qi, Weijian Li, Zhengyuan Yang, Yunhong Wang, and Jiebo Luo. Attentive relational networks for mapping images to scene graphs. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[204] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. SQuAD: 100,000+ questions for machine comprehension of text. In *The 2016 Conference on Empirical Methods in Natural Language Processing*, 2016.

[205] 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.

[206] V. Ramalingam, B. Palaniappan, N. Panchanatham, and S. Palanivel. Measuring advertisement effectiveness—a neural network approach. *Expert Systems with Applications*, 31(1):159 – 163, 2006.

[207] Anurag Ranjan and Michael J Black. Optical flow estimation using a spatial pyramid network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 2, 2017.

[208] Zeeshan Rasheed and Mubarak Shah. Movie genre classification by exploiting audio-visual features of previews. In *ICPR*, 2002.

[209] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[210] Mengye Ren and Richard S. Zemel. End-to-end instance segmentation with recurrent attention. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[211] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015.

[212] Zhou Ren, Hailin Jin, Zhe Lin, Chen Fang, and Alan Yuille. Joint image-text representation by gaussian visual-semantic embedding. In *Proceedings of the 2016 ACM on Multimedia Conference*, pages 207–211. ACM, 2016.

[213] 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)*, October 2016.

[214] Andrew Rosenberg and Julia Hirschberg. V-measure: A conditional entropy-based external cluster evaluation measure. In *EMNLP-CoNLL*, volume 7, pages 410–420, 2007.

[215] 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. *International Journal of Computer Vision*, 115(3):211–252, 2015.

[216] Juan M Sánchez, Xavier Binefa, and Jordi Vitrià. Shot partitioning based recognition of tv commercials. *Multimedia Tools and Applications*, 18(3):233–247, 2002.

[217] Dan Schill. The visual image and the political image: A review of visual communication research in the field of political communication. *Review of Communication*, 12(2):118–142, 2012.

[218] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

[219] Sebastian Schuster, Ranjay Krishna, Angel Chang, Li Fei-Fei, and Christopher D. Manning. Generating semantically precise scene graphs from textual descriptions for improved image retrieval. In *Workshop on Vision and Language (VL15)*, September 2015.

[220] Idan Schwartz, Seunghak Yu, Tamir Hazan, and Alexander G. Schwing. Factor graph attention. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[221] Linda M Scott. Images in advertising: The need for a theory of visual rhetoric. *Journal of consumer research*, 21(2):252–273, 1994.

[222] Linda M Scott and Rajeev Batra. *Persuasive imagery: A consumer response perspective*. Routledge, 2003.

[223] Arda Senocak, Tae-Hyun Oh, Junsik Kim, Ming-Hsuan Yang, and In So Kweon. Learning to localize sound source in visual scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[224] Ricky J Sethi, Yolanda Gil, Hyunjoon Jo, and Andrew Philpot. Large-scale multimedia content analysis using scientific workflows. In *Proceedings of the ACM International Conference on Multimedia*, 2013.

[225] Sanket Shah, Anand Mishra, Naganand Yadati, and Partha Pratim Talukdar. Kvqa: Knowledge-aware visual question answering. In *The Thirty-Third AAAI Conference on Artificial Intelligence*, 2019.

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

[227] 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 *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[228] Kevin J. Shih, Saurabh Singh, and Derek Hoiem. Where to look: Focus regions for visual question answering. In *Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2016.

[229] Jamie Shotton, Ross Girshick, Andrew Fitzgibbon, Toby Sharp, Mat Cook, Mark Finocchio, Richard Moore, Pushmeet Kohli, Antonio Criminisi, Alex Kipman, et al. Efficient human pose estimation from single depth images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(12):2821–2840, 2013.

[230] Zheng Shou, Hang Gao, Lei Zhang, Kazuyuki Miyazawa, and Shih-Fu Chang. Autoloc: Weakly-supervised temporal action localization in untrimmed videos. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018.

[231] Zheng Shou, Dongang Wang, and Shih-Fu Chang. Temporal action localization in untrimmed videos via multi-stage cnns. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[232] Ajeet Kumar Singh, Anand Mishra, Shashank Shekhar, and Anirban Chakraborty. From strings to things: Knowledge-enabled vqa model that can read and reason. In *The IEEE International Conference on Computer Vision (ICCV)*, 2019.

[233] Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh, and Marcus Rohrbach. Towards vqa models that can read. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[234] Surendra Singh and Catherine Cole. The effects of length, content, and repetition on television commercial effectiveness. *Journal of Marketing Research*, 30:91–104, 02 1993.

[235] Surendra N. Singh and Gilbert A. Churchill Jr. Arousal and advertising effectiveness. *Journal of Advertising*, 16(1):4–40, 1987.

[236] Khurram Soomro, Haroon Idrees, and Mubarak Shah. Action localization in videos through context walk. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, December 2015.

[237] Nancy E Spears, John C Mowen, and Goutam Chakraborty. Symbolic role of animals in print advertising: Content analysis and conceptual development. *Journal of Business Research*, 37(2):87–95, 1996.

[238] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(56):1929–1958, 2014.

[239] Horst Stipp. How context can make advertising more effective. *Journal of Advertising Research*, 58(2):138–145, 2018.

[240] Weijie Su, Xizhou Zhu, Yue Cao, Bin Li, Lewei Lu, Furu Wei, and Jifeng Dai. Vl-bert: Pre-training of generic visual-linguistic representations. *arXiv preprint arXiv:1908.08530*, 2019.

[241] Deqing Sun, Stefan Roth, and Michael J Black. Secrets of optical flow estimation and their principles. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2432–2439. IEEE, 2010.

[242] Dídac Surís, Dave Epstein, Heng Ji, Shih-Fu Chang, and Carl Vondrick. Learning to learn words from visual scenes. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.

[243] Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, and Alexander A. Alemi. Inception-v4, inception-resnet and the impact of residual connections on learning. In *The Thirty-First AAAI Conference on Artificial Intelligence (AAAI)*, 2017.

[244] 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 (CVPR)*, June 2016.

[245] Hao Tan and Mohit Bansal. Lxmert: Learning cross-modality encoder representations from transformers. In *The 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 2019.

[246] Kevin Tang, Vignesh Ramanathan, Li Fei-Fei, and Daphne Koller. Shifting weights: Adapting object detectors from image to video. In *Advances in Neural Information Processing Systems*, pages 638–646, 2012.

[247] Peng Tang, Xinggang Wang, Song Bai, Wei Shen, Xiang Bai, Wenyu Liu, and Alan Loddon Yuille. Pcl: Proposal cluster learning for weakly supervised object detection. *IEEE transactions on pattern analysis and machine intelligence*, 2018.

[248] Peng Tang, Xinggang Wang, Xiang Bai, and Wenyu Liu. Multiple instance detection network with online instance classifier refinement. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[249] Makarand Tapaswi, Yukun Zhu, Rainer Stiefelhagen, Antonio Torralba, Raquel Urtasun, and Sanja Fidler. Movieqa: Understanding stories in movies through question-

answering. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[250] Thales Teixeira, Michel Wedel, and Rik Pieters. Emotion-induced engagement in internet video advertisements. *Journal of Marketing Research*, 49(2):144–159, 2012.

[251] Damien Teney, Peter Anderson, Xiaodong He, and Anton van den Hengel. Tips and tricks for visual question answering: Learnings from the 2017 challenge. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[252] Chris Thomas and Adriana Kovashka. Persuasive faces: Generating faces in advertisements. In *Proceedings of the British Machine Vision Conference (BMVC)*, 2018.

[253] Christopher Thomas and Adriana Kovashka. Artistic object recognition by unsupervised style adaptation. In *Asian Conference on Computer Vision (ACCV)*, pages 460–476. Springer, 2018.

[254] Christopher Thomas and Adriana Kovashka. Matching complementary images and text through diversity, discrepancy and density weighting. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020. Under review.

[255] Alexander Toshev and Christian Szegedy. Deeppose: Human pose estimation via deep neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1653–1660, 2014.

[256] Florian Tramèr, Pascal Dupré, Gili Rusak, Giancarlo Pellegrino, and Dan Boneh. Adversarial: Defeating perceptual ad-blocking. *arXiv preprint arXiv:1811.03194*, 2018.

[257] 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 (ICCV)*, pages 4489–4497. IEEE, 2015.

[258] Naman Turakhia and Devi Parikh. Attribute dominance: What pops out? In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, December 2013.

[259] Jasper RR Uijlings, Koen EA Van De Sande, Theo Gevers, and Arnold WM Smeulders. Selective search for object recognition. *International Journal of Computer Vision*, 104(2):154–171, 2013.

[260] Ramakrishna Vedantam, Samy Bengio, Kevin Murphy, Devi Parikh, and Gal Chechik. Context-aware captions from context-agnostic supervision. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[261] Ramakrishna Vedantam, Karan Desai, Stefan Lee, Marcus Rohrbach, Dhruv Batra, and Devi Parikh. Probabilistic neural symbolic models for interpretable visual question answering. In *International Conference on Machine Learning*, 2019.

[262] Subhashini Venugopalan, Lisa Anne Hendricks, Marcus Rohrbach, Raymond Mooney, Trevor Darrell, and Kate Saenko. Captioning images with diverse objects. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[263] Paul Vicol, Makarand Tapaswi, Lluis Castrejon, and Sanja Fidler. Moviegraphs: Towards understanding human-centric situations from videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8581–8590, 2018.

[264] Jesse Vig. A multiscale visualization of attention in the transformer model. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 37–42, Florence, Italy, July 2019. Association for Computational Linguistics.

[265] 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*, pages 3156–3164, 2015.

[266] Fang Wan, Pengxu Wei, Jianbin Jiao, Zhenjun Han, and Qixiang Ye. Min-entropy latent model for weakly supervised object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[267] Chengyi Wang, Yu Wu, Shujie Liu, Ming Zhou, and Zhenglu Yang. Curriculum pretraining for end-to-end speech translation. *arXiv preprint arXiv:2004.10093*, 2020.

[268] Limin Wang, Yu Qiao, and Xiaoou Tang. Action recognition with trajectory-pooled deep-convolutional descriptors. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4305–4314, 2015.

[269] Limin Wang, Yuanjun Xiong, Dahua Lin, and Luc Van Gool. Untrimmednets for weakly supervised action recognition and detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[270] Peng Wang, Qi Wu, Chunhua Shen, Anthony Dick, and Anton Van Den Henge. Explicit knowledge-based reasoning for visual question answering. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, 2017.

[271] Peng Wang, Qi Wu, Chunhua Shen, Anthony Dick, and Anton van den Hengel. Fvqa: Fact-based visual question answering. *IEEE Transactions of Pattern Analysis and Machine Intelligence*, 2018.

[272] Yicheng Wang and Mohit Bansal. Robust machine comprehension models via adversarial training. In *The 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2018.

[273] Yu Wang, Yang Feng, Zhe Hong, Ryan Berger, and Jiebo Luo. How polarized have we become? a multimodal classification of trump followers and clinton followers. In *International Conference on Social Informatics*, pages 440–456. Springer, 2017.

[274] Yu-Siang Wang, Chenxi Liu, Xiaohui Zeng, and Alan Yuille. Scene graph parsing as dependency parsing. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, June 2018.

[275] Yunchao Wei, Zhiqiang Shen, Bowen Cheng, Honghui Shi, Jinjun Xiong, Jiashi Feng, and Thomas Huang. Ts2c: Tight box mining with surrounding segmentation context for weakly supervised object detection. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018.

[276] Philippe Weinzaepfel, Zaid Harchaoui, and Cordelia Schmid. Learning to track for spatio-temporal action localization. In *The IEEE International Conference on Computer Vision (ICCV)*, December 2015.

[277] Chung-Yi Weng, Wei-Ta Chu, and Ja-Ling Wu. Rolenet: Movie analysis from the perspective of social networks. *IEEE Transactions on Multimedia*, 11(2):256–271, 2009.

[278] Judith Williamson. *Decoding advertisements*. Marion Boyars Publishers Ltd, 1978.

[279] Donald R. Winslow. The pulitzer eddie adams didn't want, April 2011. `https://lens.blogs.nytimes.com/2011/04/19/the-pulitzer-eddie-adams-didnt-want/`.

[280] Donghyeon Won, Zachary C Steinert-Threlkeld, and Jungseock Joo. Protest activity detection and perceived violence estimation from social media images. In *Proceedings of the 2017 ACM on Multimedia Conference*, pages 786–794. ACM, 2017.

[281] Chao-Yuan Wu, R. Manmatha, Alexander J. Smola, and Philipp Krahenbuhl. Sampling matters in deep embedding learning. In *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[282] Hao Wu, Jiayuan Mao, Yufeng Zhang, Yuning Jiang, Lei Li, Weiwei Sun, and Wei-Ying Ma. Unified visual-semantic embeddings: Bridging vision and language with structured meaning representations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[283] Qi Wu, Peng Wang, Chunhua Shen, Anthony Dick, and Anton van den Hengel. Ask me anything: Free-form visual question answering based on knowledge from external sources. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[284] Danfei Xu, Yuke Zhu, Christopher B. Choy, and Li Fei-Fei. Scene graph generation by iterative message passing. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[285] Huijuan Xu and Kate Saenko. Ask, attend and answer: Exploring question-guided spatial attention for visual question answering. In *European Conference on Computer Vision*, pages 451–466. Springer, 2016.

[286] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhutdinov, Richard S Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention. *arXiv preprint arXiv:1502.03044*, 2(3):5, 2015.

[287] Xin Xu, Robert L Alexander, Sean A Simpson, Scott Goates, James M Nonnemaker, Kevin C Davis, and Tim McAfee. A cost-effectiveness analysis of the first federally funded antismoking campaign. *American journal of preventive medicine*, 48(3):318–325, 2015.

[288] Karthik Yadati, Harish Katti, and Mohan Kankanhalli. Cavva: Computational affective video-in-video advertising. *IEEE Transactions on Multimedia*, 16(1):15–23, 2014.

[289] Jianwei Yang, Jiasen Lu, Stefan Lee, Dhruv Batra, and Devi Parikh. Graph r-cnn for scene graph generation. In *The European Conference on Computer Vision (ECCV)*, 2018.

[290] Linjie Yang, Kevin Tang, Jianchao Yang, and Li-Jia Li. Dense captioning with joint inference and visual context. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[291] Zichao Yang, Xiaodong He, Jianfeng Gao, Li Deng, and Alex Smola. Stacked attention networks for image question answering. In *Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2016.

[292] Alexander Yates, Michele Banko, Matthew Broadhead, Michael Cafarella, Oren Etzioni, and Stephen Soderland. TextRunner: Open information extraction on the web. In *Proceedings of the Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, April 2007.

[293] Keren Ye, Kyle Buettner, and Adriana Kovashka. Story understanding in video advertisements. In *British Machine Vision Conference 2018, BMVC 2018, Newcastle, UK, September 3-6, 2018*, page 57. BMVA Press, 2018.

[294] Keren Ye and Adriana Kovashka. Advise: Symbolism and external knowledge for decoding advertisements. In *The European Conference on Computer Vision (ECCV)*, September 2018.

[295] Keren Ye and Adriana Kovashka. A case study of the shortcut effectsin visual commonsense reasoning. In *The Thirty-Fifth AAAI Conference on Artificial Intelligence (AAAI)*, February 2021.

[296] Keren Ye and Adriana Kovashka. Linguistic structures as weak supervision for visual scene graph generation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2021.

[297] Keren Ye, Narges Honarvar Nazari, James Hahn, Zaeem Hussain, Mingda Zhang, and Adriana Kovashka. Interpreting the rhetoric of visual advertisements. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2019.

[298] Keren Ye, Mingda Zhang, and Adriana Kovashka. Breaking shortcuts by masking for robust visual reasoning. In *The IEEE/CVF Winter Conference on Applications of Computer Vision*, February 2021.

[299] Keren Ye, Mingda Zhang, Adriana Kovashka, Wei Li, Danfeng Qin, and Jesse Berent. Cap2det: Learning to amplify weak caption supervision for object detection. In *The IEEE International Conference on Computer Vision (ICCV)*, October 2019.

[300] Serena Yeung, Olga Russakovsky, Greg Mori, and Li Fei-Fei. End-to-end learning of action detection from frame glimpses in videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[301] Kexin Yi, Jiajun Wu, Chuang Gan, Antonio Torralba, Pushmeet Kohli, and Josh Tenenbaum. Neural-symbolic vqa: Disentangling reasoning from vision and language understanding. In *Advances in Neural Information Processing Systems*, pages 1031–1042, 2018.

[302] Quanzeng You, Hailin Jin, Zhaowen Wang, Chen Fang, and Jiebo Luo. Image captioning with semantic attention. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[303] Charles E Young. *The advertising research handbook*. Ideas in Flight, 2005.

[304] Charles E Young. *The advertising research handbook*. Ideas in Flight, 2008.

[305] Peter Young, Alice Lai, Micah Hodosh, and Julia Hockenmaier. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the Association for Computational Linguistics*, 2:67–78, 2014.

[306] Fei Yu, Jiji Tang, Weichong Yin, Yu Sun, Hao Tian, Hua Wu, and Haifeng Wang. Ernie-vil: Knowledge enhanced vision-language representations through scene graph. *arXiv preprint arXiv:2006.16934*, 2020.

[307] Licheng Yu, Eunbyung Park, Alexander C. Berg, and Tamara L. Berg. Visual madlibs: Fill in the blank description generation and question answering. In *The IEEE International Conference on Computer Vision (ICCV)*, December 2015.

[308] Weijiang Yu, Jingwen Zhou, Weihao Yu, Xiaodan Liang, and Nong Xiao. Heterogeneous graph learning for visual commonsense reasoning. In *Advances in Neural Information Processing Systems 32*, pages 2769–2779. Curran Associates, Inc., 2019.

[309] Youngjae Yu, Jongwook Choi, Yeonhwa Kim, Kyung Yoo, Sang-Hun Lee, and Gunhee Kim. Supervising neural attention models for video captioning by human gaze data.

In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[310] Youngjae Yu, Hyungjin Ko, Jongwook Choi, and Gunhee Kim. End-to-end concept word detection for video captioning, retrieval, and question answering. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[311] Amir Zadeh, Michael Chan, Paul Pu Liang, Edmund Tong, and Louis-Philippe Morency. Social-iq: A question answering benchmark for artificial social intelligence. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[312] Amir R Zamir, Te-Lin Wu, Lin Sun, William B Shen, Bertram E Shi, Jitendra Malik, and Silvio Savarese. Feedback networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1808–1817. IEEE, 2017.

[313] Alireza Zareian, Svebor Karaman, and Shih-Fu Chang. Weakly supervised visual semantic parsing. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[314] Rowan Zellers, Yonatan Bisk, Ali Farhadi, and Yejin Choi. From recognition to cognition: Visual commonsense reasoning. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[315] Rowan Zellers, Mark Yatskar, Sam Thomson, and Yejin Choi. Neural motifs: Scene graph parsing with global context. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[316] Hanwang Zhang, Zawlin Kyaw, Shih-Fu Chang, and Tat-Seng Chua. Visual translation embedding network for visual relation detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[317] Hanwang Zhang, Zawlin Kyaw, Jinyang Yu, and Shih-Fu Chang. Ppr-fcn: Weakly supervised visual relation detection via parallel pairwise r-fcn. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[318] Mingda Zhang, Rebecca Hwa, and Adriana Kovashka. Equal but not the same: Understanding the implicit relationship between persuasive images and text. In *British Machine Vision Conference (BMVC)*, 2018.

[319] Peng Zhang, Yash Goyal, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Yin and yang: Balancing and answering binary visual questions. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[320] Yang Zhang, Philip David, and Boqing Gong. Curriculum domain adaptation for semantic segmentation of urban scenes. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 2020–2030, 2017.

[321] Fang Zhao, Jianshu Li, Jian Zhao, and Jiashi Feng. Weakly supervised phrase localization with multi-scale anchored transformer network. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[322] Yue Zhao, Yuanjun Xiong, Limin Wang, Zhirong Wu, Xiaoou Tang, and Dahua Lin. Temporal action detection with structured segment networks. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[323] Heliang Zheng, Jianlong Fu, Tao Mei, and Jiebo Luo. Learning multi-attention convolutional neural network for fine-grained image recognition. In *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[324] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[325] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database for scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017.

[326] Bolei Zhou, Yuandong Tian, Sainbayar Sukhbaatar, Arthur Szlam, and Rob Fergus. Simple baseline for visual question answering. *arXiv preprint arXiv:1512.02167*, 2015.

[327] Yuke Zhu, Oliver Groth, Michael Bernstein, and Li Fei-Fei. Visual7w: Grounded question answering in images. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[328] Yuke Zhu, Joseph J. Lim, and Li Fei-Fei. Knowledge acquisition for visual question answering via iterative querying. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[329] C Lawrence Zitnick and Piotr Dollár. Edge boxes: Locating object proposals from edges. In *European Conference on Computer Vision*, pages 391–405. Springer, 2014.