ACTION MODELING IN LONG-FORM VIDEOS

By

Junwen Chen

A DISSERTATION

Submitted to
Michigan State University
in partial fulfillment of the requirements
for the degree of

Computer Science—Doctor of Philosophy

2024

ABSTRACT

Video is the dominant modality that people use to consume content and share experiences in time series. The significant expansion of video data available both on the internet and in everyday life has spurred the creation of intelligent systems that can automatically analyze video content and comprehend human actions. Compared to static images, videos describe how the world changes as time elapses. The uniqueness of videos, beyond what can be understood from a single image, is the context of action understanding. Over the last ten years, we have seen huge success in recognizing human actions in a video, by deep neural networks. However, this action recognition has several limitations for real applications. It primarily focuses on recognizing action patterns within only a few seconds. This is still far from progressing to a human-level intelligence of video understanding. People can directly perceive uncurated long videos in the real world. We want the model directly applied to long-form videos, which are untrimmed and contain multiple actions/events.

In approaching this challenge, in this thesis, we first study the representation learning of actions/events in long-form videos. We develop models to learn the fine-grained motion representations across multiple actions/events in a video. My research seeks to enable machine visions to represent motions over a long-horizon range, by exploiting the potential of multi-modal video-language contexts. We also address learning the actions jointly performed by a group of people, by modeling their interactions. After that, we investigate leveraging the long-range dependencies of the events in boosting temporal reasoning downstream tasks, including online action detection and spatiotemporal object grounding. Finally, considering the wide applications of video models, we focus on cultivating trustworthiness in the models for long-form videos from static bias mitigation and interpretable reasoning perspectives.

ACKNOWLEDGEMENTS

I would like to express my sincere thanks to all the people who helped and supported me in this invaluable academic discovery journey. First of all, I am full of appreciation for my advisor Professor Yu Kong who has taken me under his wing since the beginning of his faculty career and supported me all these years. Thank you for the days and nights we worked together, discussing research ideas, sharing insightful work, writing papers and responses, and sharing a cake when our effort paid off. He offered me the space to grow and the great freedom to explore the topics that I am interested in. He also gave me a lot of courage and recognition to pose new research questions and tackle challenging and exciting research problems. When I encountered difficulties, he provided me with constructive suggestions and cheered me up. Besides, I am sincerely thankful to Dr. Kong and his family who overcame many obstacles moving to MSU, so that I could have the opportunity to experience two different cities and departments and meet a diversity of researchers.

I am deeply grateful to my committee members, Prof. Xiaoming Liu, Prof. Parisa Kordjamshidi, and Prof. Daniel Morris, all of whom have given me invaluable support and provided precious suggestions and feedback on my research and thesis. Their students have also been intellectually stimulating and socially fun to me. I would extend gratitude to my internship mentors as well.

I would like to thank our Action lab: When I first joined the lab, I had accumulated less research experiences in computer vision compared to Wentao. He spent countless efforts in helping me present results, polish papers, and write responses, even when he had deadlines to rush. I am happy and grateful to have the young generation Yujiang, Yifan, and Zhanbo as labmates in the final year of my Ph.D., with whom I shared common research interests and got a lot of motivations. Many thanks also to the former members, Haiting, Hanbin, Chuanqi, Anna, Xinmiao, and Xiwen.

During my Ph.D., I am fortunate to spend unforgettable four years in Rochester, the birthplace of Kodak, where I met researchers working on a variety of computer vision and imaging related projects, including Prof. Linwei Wang (the role model to a female Ph.D. student), Prof. Pengcheng Shi, Prof. Rui Li, Prof. Matthew Wright, Prof. Dongfang Liu, Prof. Chenliang Xu and his group.

Finally, I would like to thank my parents and family, whose endless love serves as my cornerstone.

TABLE OF CONTENTS

CHAPTER 1 INTRODUCTION 1

BIBLIOGRAPHY 7

CHAPTER 2 LEARNING FINE-GRAINED MOTION REPRESENTATIONS 9

BIBLIOGRAPHY 27

CHAPTER 3 GROUP ACTIVITY PREDICTION WITH SEQUENTIALLY RELATIONAL ANTICIPATION MODEL 32

BIBLIOGRAPHY 50

CHAPTER 4 GATED HISTORY UNIT WITH BACKGROUND SUPPRESSION FOR ONLINE ACTION DETECTION 54

BIBLIOGRAPHY 72

CHAPTER 5 ACTIVITY-DRIVEN WEAKLY-SUPERVISED SPATIAL-TEMPORAL VIDEO OBJECT GROUNDING 76

BIBLIOGRAPHY 97

CHAPTER 6 EXPLAINABLE VIDEO ENTAILMENT WITH VISUALLY GROUNDED EVIDENCE 100

BIBLIOGRAPHY 117

CHAPTER 7 CONCLUSIONS AND FUTURE WORK 120

CHAPTER 1

INTRODUCTION

1.1 Motivating Problems

Video is the primary signal that we perceive the world every day, as we observe our surrounding environment in the form of continuous visual input. We are in the era of video. In the real world, there are more than 45 billion cameras on the Earth now. On the Internet, hundreds of hours of videos are uploaded to video-sharing platforms such as YouTube and TikTok every single minute. One of the fundamental goals of AI research is to equip intelligent systems to analyze video content automatically and perceive their environment. For example, video-sharing websites need to understand the characters and events to make a promising recommendation; autonomous vehicle needs to predict whether a pedestrian will cross the street or not at a stop sign, to predict what he may perform next.

A video task uniquely suited for videos, beyond what can be understood from a still image (e.g. scenes, people, and objects), is the context of action understanding in videos. Human action encodes how an actor's relationship with surroundings evolves over time. Over the last ten years, we have seen big success in recognizing human actions in a video, by deep neural networks Bertasius et al. (2021); Feichtenhofer et al. (2019); Carreira and Zisserman (2017). We now have visual systems that can accurately recognize human actions from various perspectives: in different camera angles from third-person videos Carreira and Zisserman (2017); Shao et al. (2020) to first-person videos Damen et al. (2018); Grauman et al. (2022) by wearable cameras, in different granularity from coarse-level Carreira and Zisserman (2017) to fine-level Shao et al. (2020), in different modalities from pixel Carreira and Zisserman (2017); Shao et al. (2020) to human pose Shahroudy et al. (2016), and in a different number of actors from simple limb movement Shahroudy et al. (2016) to group activities of multiple actors Ibrahim et al. (2016). Powered by the ubiquitous large-scale vision(-language) models Xu et al. (2021); Radford et al. (2021); Dosovitskiy et al. (2021), these models have demonstrated near human-level performance on the recognition tasks Bertasius et al. (2021); Wang et al. (2016); Feichtenhofer et al. (2019) and

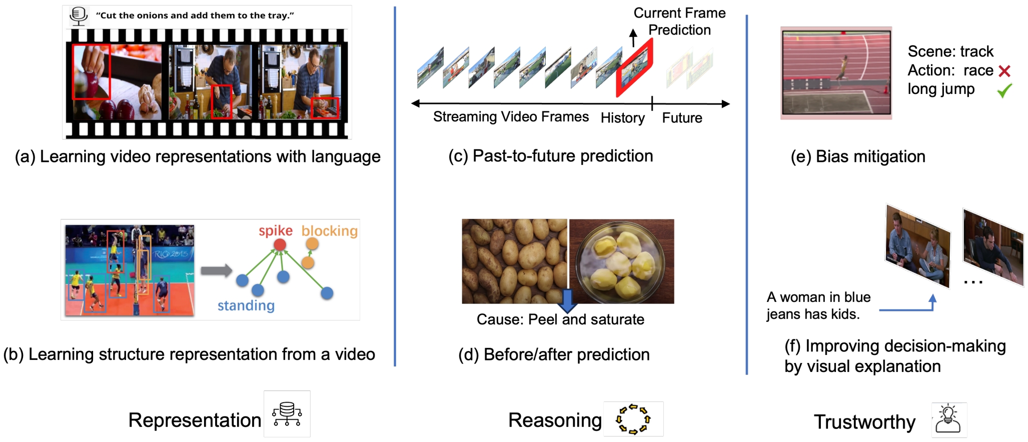

"Cut the onions and add them to the tray."
(a) Learning video representations with language
spike
blocking
standing
(b) Learning structure representation from a video
Current Frame Prediction
Streaming Video Frames
History
Future
(c) Past-to-future prediction
Cause: Peel and saturate
(d) Before/after prediction
Scene: track
Action: race ✗
long jump ✓
(e) Bias mitigation
A woman in blue jeans has kids.
...
(f) Improving decision-making by visual explanation
Representation
Reasoning
Trustworthy

Figure 1.1 Action Modeling in Long-form Videos: Representations, Reasoning, and Trustworthiness.

localization tasks Dai et al. (2017); Zhao et al. (2017). However, this action recognition has several limitations for real applications, since it primarily focuses on recognizing action patterns within only a few seconds. In the real world, we desire the model directly applied to raw untrimmed videos which may be prolonged and contain multiple events across time. To realize the goal, we need to study action modeling in long-form video understanding.

In this dissertation, we study three problems that play important roles in long-form videos (See Fig. 1.1): (1) *How to perceive the actions/events from long-form videos?* Action modeling is a fundamental task in video understanding. Challenges of action modeling from long-form videos stem from learning the fine-grained motion representations across multiple events performed on the individuals and learning the actions performed by multiple people. (2) *How to capture the long-range dependencies across time and conduct temporal reasoning between the events?* An untrimmed video usually contains rich temporal dependencies in time series. Because a baby fell down seconds ago, the baby is crying now. In addition to progressing in the perception of actions, it is necessary to design methods that learn the temporal relations in a video, such as predicting what is about to happen and why something is happening. (3) *How to cultivate trustworthiness in video models?* The diverse video understanding tasks and applications also call for designing trustworthy models. Specifically, we study the bias mitigation in video understanding and present explainable decision-making for humans.

1.2 Our Approaches

1.2.1 Learning action representations from long-form videos

In the first part of the thesis, we address the research question on *How to perceive the actions/events from long-form videos*. Existing action recognition tasks mainly benchmark the action recognition in videos with only a few seconds. We focus on learning the representations of dynamics in long-form videos, where multiple actions are performed. In the first work, we leverage video-language alignment to learn the fine-grained motions across multiple events in the same scenario. In the second work, we present a novel framework to learn the dynamics performed by multiple actors.

Specifically, in Chapter 2, we take video question answering (VideoQA) datasets that feature temporal reasoning to evaluate the fine-grained motion representations. For example, imagine that answer the question "What did the boy do before he raised his hand to take the camera?". The model needs to recognize the actions of "raising hand" and "taking camera" in the video, which are performed by the same individual "the boy" sequentially and share the same appearance information. Thus, we approach a fine-grained motion representation learning for this challenge. We introduce Action Temporality Modeling (ATM) via three-fold uniqueness: (1) an empirical study of realizing that optical flow is effective in capturing the long horizon temporality reasoning; (2) training the visual-text embedding by contrastive learning in an action-centric manner, leading to better action representations in both vision and text modalities; and (3) preventing the model from answering the question given the shuffled video in the fine-tuning stage, to avoid spurious correlation between appearance and motion and hence ensure faithful temporality reasoning. In the experiments, we show that ATM outperforms previous approaches in terms of accuracy on multiple VideoQAs and exhibits better true temporality reasoning ability.

In Chapter 3, we further investigate the representation learning of actions that are jointly performed by multiple actors, which has applications in many surveillance scenarios. We propose a novel approach to predict group activities given the beginning frames with incomplete activity executions. For group activity prediction, the relation evolution of people's activity and their positions over time is an important cue for predicting group activity. To this end, we propose a sequential relational anticipation model (SRAM) that summarizes the relational dynamics in the partial observation and progressively anticipates the group representations with rich discriminative information. Our model explicitly anticipates both activity features and positions by two graph auto-encoders, aiming to learn a discriminative group representation for group activity prediction. Experimental results on two popularly used datasets demonstrate that our approach significantly outperforms state-of-the-art action prediction methods.

1.2.2 Leveraging long-range dependencies in long-form video understanding

After extracting the representation of the individual actions/events in a long-form video, in the second part of the thesis, we further study *i.e.,* the long-range dependencies of the actions and leverage them in downstream tasks. First, we develop a vision algorithm that is capable of having a past-to-future reasoning in a fluid video stream. The algorithm can efficiently detect the relevant information from the long and redundant history frames. Second, we improve the spatiotemporal grounding of the objects by modeling the effect of human actions.

In Chapter 4, the research strives to address how to relate the long and redundant history to understanding the present. Online action detection is the task of predicting the action as soon as it happens in a streaming video. A major challenge is that the model does not have access to the future and has to rely solely on history, *i.e.,* , the frames observed so far, to make predictions. It is therefore important to accentuate parts of the history that are more informative to the prediction of the current frame. We present Gated History Unit with Background Suppression *i.e.,* GateHUB, that comprises a novel gated cross-attention mechanism to enhance or suppress parts of the history as per how informative they are for current frame prediction. In a single unified framework, GateHUB integrates the transformer's ability of long-range temporal modeling and the recurrent model's capacity to selectively encode relevant information. Extensive validation demonstrates that GateHUB significantly outperforms all existing methods and is also more efficient than the existing best work.

When a human performs action, the carrier of action *i.e.,* , objects experiences the state change, which is considered the effect of human action. For example, when doing “mash potato”, we can see the potato evolve from “cube-shape” to “paste-shape”. Given our dynamic world, I contend that traditional object-centric tasks such as grounding can be facilitated with videos, where the cause-effect is an intrinsic cue for learning to see. Chapter 5 pointed out that activity cues in both text and visual modalities are informative for grounding objects in an untrimmed video. This work is one of the first to combine the cause-effect from instructional videos, which is a setting that is more likely to become common as environmental and wearable cameras become even more ubiquitous.

1.2.3 Cultivating trustworthiness in video models

While video models can be widely adopted in many tasks and daily applications, it is necessary to guarantee that the developed models are trustworthy, especially the human-centered visual tasks. In the third part of the thesis, we work on making model decision-making trustworthy, by (1) revealing and mitigating the bias in video understanding and (2) making the decision-making interpretable, both leveraging the multimodal context.

Recent studies have pointed out that the models sometimes capture the unwanted bias exhibited in training data. In addition to the widely considered bias *e.g.,* , gender, race, and watermark, videos contain a specific type of bias, *i.e.,* static bias. That is exploiting the static representations (objects, scenes, and people) to learn the underlying temporal reasoning task. For example, while the “basketball dunk” and “soccer juggling” have distinct temporal patterns, they can be discriminated by classifying the background into a basketball court or a soccer field. In Chapter 2, the proposed ATM also contributes to overcoming static bias in temporal reasoning by manipulating text supervision. Particularly, we design a simple yet effective solution that masks the appearance word in the text and guides the video representation to be aligned with the motion word. We also design a new metric to reveal if the temporal reasoning question answering relies on the actual motion information.

In Chapter 6, we propose to improve the video language reasoning task performance with visually grounded evidence. In addition to the downstream task reasoning part, we add the explanation part that grounds the people and objects mentioned in the text to the videos. We leverage video entailment as evaluation, which aims at determining if a hypothesis textual statement is entailed or contradicted by a premise video. The main challenge of video entailment is that it requires fine-grained reasoning to understand complex and long story-based videos. We proposes to incorporate visual grounding to the entailment by explicitly linking the entities described in the statement to the evidence in the video. If the entities are grounded in the video, we enhance the entailment judgment by focusing on the frames where the entities occur. Besides, in the entailment dataset, the entailed/contradictory (also named as real/fake) statements are formed in pairs with the subtle discrepancy, which allows an add-on explanation module to predict which words or phrases make the statement contradictory to the video and regularize the training of the entailment judgment. Experimental results demonstrate that our approach outperforms the state-of-the-art methods.

1.3 Relevant Publications

- Chapter 2- Junwen Chen, Jie Zhu, and Yu Kong. ATM: Action Temporality Modeling for Video Question Answering. In ACM Multimedia, 2023

- Chapter 3- Junwen Chen, Wentao Bao, and Yu Kong. Group Activity Prediction with Sequential Relational Anticipation Model. In European Conference on Computer Vision (ECCV), 2020

- Chapter 4- Junwen Chen, Gaurav Mittal, Ye Yu, Yu Kong, and Mei Chen. GateHUB: Gated History Unit with Background Suppression for Online Action Detection. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2022

- Chapter 5- Junwen Chen, Wentao Bao, and Yu Kong. Activity-driven Weakly-Supervised Spatio-Temporal Grounding from Untrimmed Videos. In ACM Multimedia, 2020

- Chapter 6- Junwen Chen and Yu Kong. Explainable Video Entailment with Visually Grounded Evidence. In IEEE International Conference on Computer Vision (ICCV), 2021.

BIBLIOGRAPHY

Bertasius, G., Wang, H., and Torresani, L. (2021). Is space-time attention all you need for video understanding?

Carreira, J. and Zisserman, A. (2017). Quo vadis, action recognition? A new model and the Kinetics dataset. In *CVPR*.

Dai, X., Singh, B., Zhang, G., Davis, L. S., and Chen, Y. Q. (2017). Temporal context network for activity localization in videos. In *ICCV*.

Damen, D., Doughty, H., Farinella, G. M., Fidler, S., Furnari, A., Kazakos, E., Moltisanti, D., Munro, J., Perrett, T., Price, W., and Wray, M. (2018). Scaling egocentric vision: The epic-kitchens dataset. In *European Conference on Computer Vision (ECCV)*.

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

Feichtenhofer, C., Fan, H., Malik, J., and He, K. (2019). Slowfast networks for video recognition. In *ICCV*.

Grauman, K., Westbury, A., Byrne, E., Chavis, Z., Furnari, A., Girdhar, R., Hamburger, J., Jiang, H., Liu, M., Liu, X., et al. (2022). Ego4d: Around the world in 3,000 hours of egocentric video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18995–19012.

Ibrahim, M. S., Muralidharan, S., Deng, Z., Vahdat, A., and Mori, G. (2016). A hierarchical deep temporal model for group activity recognition. In *CVPR*, pages 1971–1980.

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

Shahroudy, A., Liu, J., Ng, T.-T., and Wang, G. (2016). Ntu rgb+d: A large scale dataset for 3d human activity analysis. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1010–1019.

Shao, D., Zhao, Y., Dai, B., and Lin, D. (2020). Finegym: A hierarchical video dataset for fine-grained action understanding. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2616–2625.

Wang, L., Xiong, Y., Wang, Z., Qiao, Y., Lin, D., Tang, X., and Van Gool, L. (2016). Temporal segment networks: Towards good practices for deep action recognition. In *ECCV*.

Xu, H., Ghosh, G., Huang, P.-Y., Okhonko, D., Aghajanyan, A., Metze, F., Zettlemoyer, L., and Feichtenhofer, C. (2021). Videoclip: Contrastive pre-training for zero-shot video-text understanding. In *EMNLP*, pages 6787–6800.

Zhao, Y., Xiong, Y., Wang, L., Wu, Z., Tang, X., and Lin, D. (2017). Temporal action detection with structured segment networks. In *ICCV*.

CHAPTER 2

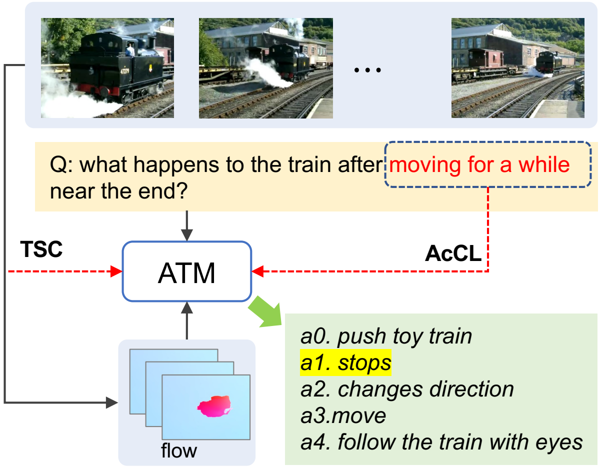
LEARNING FINE-GRAINED MOTION REPRESENTATIONS

2.1 Introduction

Video question answering (VideoQA) is an interactive AI task, which enables many downstream applications such as vision-language navigation and communication systems. It aims to answer the natural language question given the video content. Recent VideoQA benchmark Xiao et al. (2021) has gone beyond the description of video content like "*A baby is crying*" and started to provide effective diagnostics for the models on solving temporal reasoning and causal reflection, *e.g.,* "*The train stops after moving for a while*". To correctly answer the question, a VideoQA model needs to detect the object "train", recognize the "railway" scene, more importantly ground the action "move" and "stop" and understand their temporal relations. The questions are unconstrained and complex, and thus, it is necessary to have a visual-text alignment model that has the reasoning capability towards all aforementioned contents.

Recent advanced VideoQA models have shown the capability of learning from the descriptive contents Lei et al. (2018, 2019), thanks to the success of cross-modal transformers Li et al. (2020); Lei et al. (2021). However, the temporality reasoning in videos remains a great challenge, since these VideoQAs are only capable of holistic recognition of static content in a video. Recent work attempt to solve this issue by (1) enhancing the video representation with fine-grained dynamics Xiao et al. (2022b,a) and (2) answering by grounding to question-critical visual evidence Li et al. (2022b,c). But it is hard to achieve a precise grounding, without the ground-truth of temporal boundaries for training. The state-of-the-art method VGT Xiao et al. (2022b) proposes to model the atomic actions across frames from the spatio-temporal dynamics of objects. In this way, the fine-grained dynamics can be captured. But their model may rely on the static bias *i.e.,* object appearance, as shortcuts from videos while the causal factors *i.e.,* the dynamics are overlooked in training. In this paper, we address the importance of precise and faithful modeling of actions for the VideoQA task.

We propose Action Temporality Modeling (ATM) to address the challenging temporality VideoQA (as shown in Fig. 2.1). A promise of VideoQA compared to ImageQA is to examine



...
Q: what happens to the train after moving for a while near the end?
TSC
ATM
AcCL
flow
a0. push toy train
a1. stops
a2. changes direction
a3.move
a4. follow the train with eyes

Figure 2.1 ATM addresses VideoQA featuring multi-frame temporality reasoning by (1) an appearance-free stream *i.e.,* optical flow to extract precise motion cues, (2) Action-centric contrastive learning (AcCL) for an action-plentiful cross-modal motion representation, and (3) a temporal sensitivity-aware confusion (TSC) training to avoid learning a shortcut between temporality-critical motion and appearance.

the temporal relation reasoning regarding motion information. As the targeted video is continuous, actions across a long video usually share the same scene in short moments. We realize that (1) leveraging an appearance-free stream *e.g.,* optical flow as input, though the flow stream may become less considered in recent action recognition methods Bertasius et al. (2021); Feichtenhofer et al. (2019), is still important in VideoQA. Because flow can capture the subtle transition in long horizon and aid the temporality reasoning. ATM trains the visual-text encoding in a contrastive manner. Questions are usually unconstrained in the real world. Action may be only a small portion of the question, which is easily overwhelmed by other information such as objects. (2) To learn an action-plentiful cross-modal embedding, we develop a novel action-centric contrastive learning (AcCL) before fine-tuning VideoQA. Specifically, it parses an action phrase from a question and encourages a feature alignment between the video and the parsed action phrase alone, discarding other textual information. The merit of the AcCL is that both video and text encoders are trained to focus on actions, mitigating the backbone's representation bias towards the static visual appearance in videos.

Based on the learned representations, we further introduce a novel temporal sensitivity-aware confusion loss (TSC) in VideoQA finetuning. It prevents a model from answering a temporality question if the corresponding video is shuffled in the temporal domain, thus avoiding simply learning the shortcut correlation to the static content. Note that VideoQA contains a lot of descriptive questions that can be answered invariant to temporal change. Thus, we only apply the confusion loss to temporal-sensitive questions that contain temporal keywords.

Thanks to these components, the proposed ATM outperforms all of the existing methods on three commonly used VideoQA datasets. It is worth noting that our method without external vision-language pretraining can surpass the existing method that relies on large-scale pre-training by a clear margin. Moreover, we devise a new metric that quantifies the accuracy difference between conditioned on a full video and conditioned on a single frame, which reveals the VideoQA's true temporality reasoning ability. Results show that our model experiences a larger performance escalation from a single frame to a full video, which demonstrates ours relies on less appearance bias and handles temporal reasoning in a faithful manner. To summarize, our main contributions are as follows:

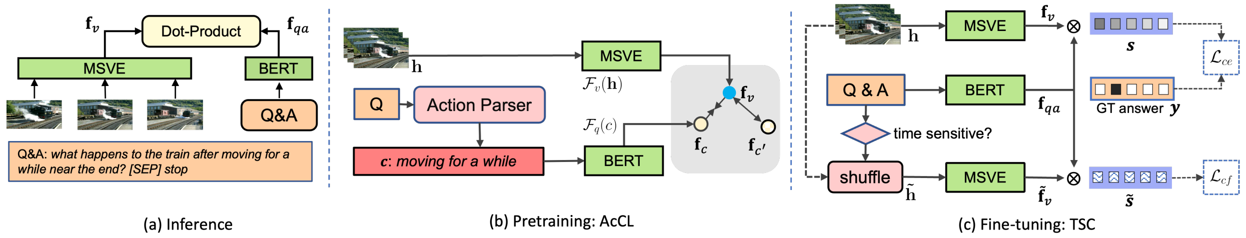
- We propose the ATM to address event temporality reasoning in VideoQA by faithful action modeling. Our action-centric contrastive learning learns action-aware representations from both vision and text modalities. We realize an appearance-free stream is effective in the multi-event temporality understanding across frames.

- We fine-tune the model with a newly developed temporal sensitivity-aware confusion loss that mitigates static bias in temporality reasoning.

- Our method is more accurate than all existing methods on three widely used VideoQA datasets. By a new metric, we also indicate that our method addresses temporality reasoning more faithfully.

2.2 Related Work

Video Question Answering. Escalating ImageQA Antol et al. (2015), VideoQA Xu et al. (2016); Yu et al. (2019); Li et al. (2016); Xiao et al. (2021); Li et al. (2022a); Lei et al. (2018)



(a) Inference

(b) Pretraining: AcCL

(c) Fine-tuning: TSC

Figure 2.2 Framework Overview. Following the recent VQAs Xiao et al. (2022b); Yang et al. (2021a), we solve VideoQA by a similarity comparison between video and text (a). To achieve this, we formulate the training procedure into two stages. Before finetuning, we present a novel action-centric contrastive learning (AcCL) to guide the visual and text representation expressive for action information (b). After that, we fine-tune the VideoQA (c) by a newly developed temporal sensitivity-aware confusion loss (TSC) to prevent leveraging static bias in temporality reasoning.

is enriched with reasoning about temporal nature. Prior arts Le et al. (2020); Park et al. (2021); Xiao et al. (2022a) on VideoQA focus on learning an informative video content representation and a cross-modal fusion model to answer the question. An informative video representation is usually hierarchical, fusing object-, frame- and clip-level representations, which are extracted by graph neural network Jiang and Han (2020); Li et al. (2022c); Park et al. (2021), relation learning or transformers. While those VideoQA methods achieve compelling results on VideoQA benchmarks, they mainly answer descriptive questions for the video content, such as questions that holistic recognize the main actions/objects across frames.

Recent benchmark Xiao et al. (2021) begins to challenge the temporal relationship reasoning ability, as actions in videos are diverse and causally dependent. Those methods that are only capable of descriptive content recognition cannot perform well, because they hardly capture the subtle transitions in the same scene. To this end, recent work Xiao et al. (2022a,b) proposes to encode video as a local-to-global dynamic graph of spatiotemporal objects, so that the interaction relations can be encoded. However, the VideoQA model built upon the dynamic graph may easily be distracted by the object's appearance and capture limited motion information. We alleviate the distraction by a novel two-stage training to ensure a faithful representation of motions that are critical for temporality reasoning.

Static Bias in Video. The promise of video lies in the potential to go beyond image-level understanding *i.e.,* scenes, objects, and people to capture the temporality of events. However, for many video(+language) tasks and datasets, given just a single frame of video, an existing image-centric model can achieve surprisingly high performance, comparable to the model using multiple frames. The strong single-frame performance suggests that the video representation is biased towards the still appearance information, namely "static appearance bias". Existing work Buch et al. (2022); Lei et al. (2022); Li et al. (2018); Choi et al. (2019) reveals this kind of bias in action recognition dataset Carreira and Zisserman (2017); Soomro et al. (2012) and retrieval dataset Liu et al. (2019); Luo et al. (2021). Circling around the fundamental video task action recognition, Li et al. (2018); Choi et al. (2019) analyze the role of temporality in action recognition and inspires the subsequent development of profound faithful evaluations Shao et al. (2020); Li et al. (2018) and model structures Feichtenhofer et al. (2019); Lin et al. (2019); Feichtenhofer (2020); Duan et al. (2022).

To address the challenging temporality reasoning in multi-modal scenarios *i.e.,* VideoQA, motion representations, unbiased toward appearance, are necessary. As VideoQA requires a deep understanding of open-vocabulary action semantics, existing VideoQAs Le et al. (2020); Xiao et al. (2022a) extract the motion features based on backbones pre-trained on a large-scale action recognition dataset Carreira and Zisserman (2017). As mentioned, static bias exists in action recognition, which makes the motion representations not the causal factors of actions, thus useless to temporality reasoning. Existing methodsChoi et al. (2019); Li et al. (2018) mitigate the static bias in action recognition by evaluating it on fine-grained action recognitionShao et al. (2020); Li et al. (2018), where the scene context is the same across the different actions. However, in fine-grained action recognition, motion is more critical information, which is different from VideoQA where object/entity appearance is inevitable.

To mitigate static bias in VideoQA, IGV Li et al. (2022c) and EIGV Li et al. (2022b) are proposed to ground the question-critical scenes across frames as the evidence of yielding the answers. However, the dominant content of a question is appearance information *e.g.,* people, objects, and locations. The grounding may pay less attention to the actions that are critical for temporality understanding and be not precise as no ground truth boundaries are provided. Our method designs two simple yet effective schemes that learn faithful visual and text representations informative for action and temporality. We also revisit the early action recognition work Wang et al. (2016); Carreira and Zisserman (2017) and enhance the motion representation with an appearance-free stream.

2.3 Methodology

Figure 2.2 gives an overview of ATM framework. Our framework addresses the VideoQA task that challenges the temporal reasoning of dynamics in a video. Following the recent VQAs Yang et al. (2021a); Xiao et al. (2022b), we solve VideoQA by a similarity comparison between video and text (Figure 2.2-a). To achieve this, we formulate the training procedure into two stages. In the first stage (Figure 2.2-b), we present a novel action-centric contrastive learning (AcCL, Sec. 2.3.3), which makes the visual and text representation lexpressive for action information. After that, we finetune the VideoQA (Figure 2.2-c) by a newly developed temporal sensitivity-aware confusion loss (TSC, Sec. 2.3.4) to prevent leveraging static bias in temporality reasoning. We detailed the video and text encoding in Sec. 2.3.2

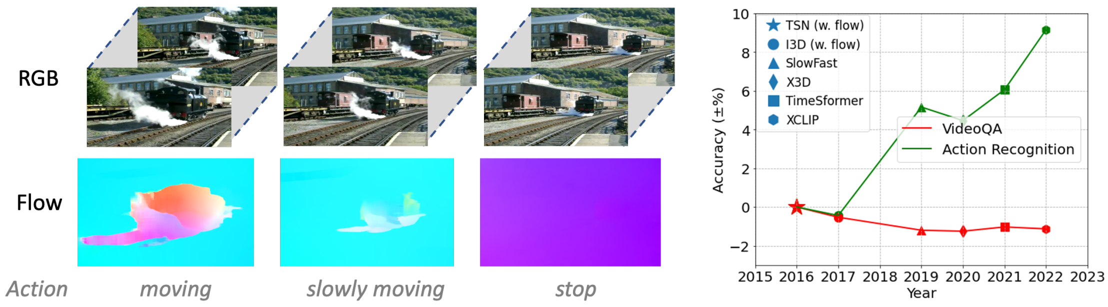
2.3.1 Preliminaries

Given a video $\mathbf{h}$ and a question q, VideoQA aims to combine the two modalities $\mathbf{h}$ and q to predict the answer a. Following existing VideoQA work Li et al. (2022c); Xiao et al. (2022a,b), we predict the answer by selecting the best matched a^* from many candidates $\mathcal{A}$ of a question q, given the corresponding video $\mathbf{h}$:

$$a^* = \arg\max_{a\in\mathcal{A}} \mathcal{F}_W(a|q, \mathbf{h}, \mathcal{A}), \tag{2.1}$$

where $\mathcal{F}_W$ denotes the mapping function with learnable parameters W. The candidates $\mathcal{A}$ are multi-choices in multi-choiceQA or a global answer list in open-ended QA.

Prior arts on VideoQA usually build $\mathcal{F}_W$ as a cross-attention transformer Zhu and Yang (2020); Lei et al. (2021), which takes a holistic token sequence containing video, question and each candidate answer as input and classifies the answers as output. Recent work VGT Xiao et al. (2022b) and VQA-T Yang et al. (2021b) propose to design $\mathcal{F}_W$ as two unimodal transformers that



(a) (b)

Figure 2.3 **Motivation of using an appearance-free stream for motion representation in VideoQA task**. The example in (a) shows the state transition on a train, from moving to stopping. We can see flow provides better cues for the actions than RGB. (b) summarizes the relative performance gain/loss of different video backbones pivot on TSN, for both action recognition (Kinetics Carreira and Zisserman (2017)) and VideoQA (NextQA Xiao et al. (2021)), which shows appearance-free stream *i.e.*, flow is necessary for VideoQA. The numbers for action recognition (green curves) are reported in their paper for Kinetics-400. The numbers for VideoQA are derived based on our implementation on NextQA.

encode video and question-answer pair respectively and compare the visual-text similarity for each answer as output:

$$s_a = \mathcal{F}_v\left(\mathbf{h}\right)\mathcal{F}_q\left([q;a]\right)^{\top}, \tag{2.2}$$

in which $\mathcal{F}_v$ denotes the video encoder and $\mathcal{F}_v\left(\mathbf{h}\right) \in \mathbb{R}^d$ is the video' global feature obtained by mean-pooling the features across T frames. Likewise, $\mathcal{F}_q$ denotes the text encoder and $\mathcal{F}_q\left([q;a]\right) \in \mathbb{R}^d$ is the feature vector of a question-answer pair, where $[;]$ indicates the concatenation of question and answer text. The visual-text similarity s_a is obtained via a dot-product of video and text features *w.r.t.* the answer a. The optimal answer is selected by maximizing the similarity score from the candidate in the pool $\mathcal{A}$:

$$a^* = \arg\max_{a\in\mathcal{A}}(s_a). \tag{2.3}$$

Following existing work Xiao et al. (2022b); Li et al. (2022c), we implement $\mathcal{F}_q$ by the BERT Devlin et al. (2019) to extract text features. For video modality, many existing methods Xiao et al. (2022b,a) extract features in multiple streams including object-level and frame-level. Following them, we also formulate $\mathcal{F}_v$ as a multi-stream video encoder (MSVE), by which object features are encoded as $f_o \in \mathbb{R}^{T\times d}$ and frame features are encoded as $f_i \in \mathbb{R}^{T\times d}$. The object/frame feature extraction and transformer-based encoding are exactly the same as state-of-the-art method VGT Xiao et al. (2022b) for a fair comparison.

2.3.2 Rethinking motion representations in VideoQA

In video feature extraction of both the existing methods Xiao et al. (2022b,a); Le et al. (2020) and ours, frame-level features $f_i \in \mathbb{R}^{T\times d}$ and object features $f_o \in \mathbb{R}^{T\times d}$ both represent appearance. Optionally, they Xiao et al. (2022a); Le et al. (2020) apply a pre-trained 3D Conv network Carreira and Zisserman (2017) on the neighboring frames to capture motions. However, VideoQA studies the temporality of the actions in a video where multiple actions are performed across frames. As a video captures continuous information, these actions usually share the same scene context and are performed by the same people and on the entity. In this case, although 3D Conv can capture motions, neighboring RGB frames may be too redundant to precisely model the actions. For example, in Figure 2.3-a, it is hard to recognize "the train is stopping" in the last clip from RGB. This inspires us to enhance the video representation by a stream, where the appearance information is least and hence the motions are highlighted. To this end, we resort to optical flow that describes the apparent motion of individual pixels on the image. As shown in Figure 2.3-a's example, flow maps provide better cues to understand the state transition of objects *e.g.,* "train" was moving (in the first and second clip) and stopped (in the third clip).

As VideoQA requires the open-vocabulary semantic understanding of motions, we use the backbone pretrained on a large-scale action recognition dataset Kinetics-400 Kay et al. (2017) to extract flow features. Flow features are extracted as per appearance frame timestamps as $f_m \in \mathbb{R}^{T\times d}$. To fuse the object, appearance, and flow streams, our MSVE applies MLPs and a learnable multi-head self-attention layer MSA with position embedding to model the temporal interactions upon the multi-stream features and mean-pool the frames to obtain the global video representation f_v.

$$f_v = \text{Mean-Pool}(\text{MSA}(\text{MLP}([f_o; f_r; f_m]))) \tag{2.4}$$

Note that we should not ignore the appearance information in VideoQA task, as the questions are unconstrained and may contain characters, objects and locations that need to be grounded to videos. This is different from the action segmentation Ding and Yao (2021) or skeleton-based activity recognition Zhou et al. (2021); Duan et al. (2022), where motion is the only critical information.

We revisit the fundamental video understanding task *i.e.,* action recognition, in which the early methods *e.g.,* TSN Wang et al. (2016); Carreira and Zisserman (2017) also utilized optical flow to capture motions. As shown in Figure 2.3-b, we observe that although the existing powerful backbones e.g. SlowFast Feichtenhofer et al. (2019), X3D Feichtenhofer (2020), TimeSformer Bertasius et al. (2021) and XCLIP Ni et al. (2022) achieve good performance w/o appearance-free stream *i.e.,* optical flow in action recognition, they are less helpful in VideoQA compared to the early methods w/ appearance-free stream. This demonstrates that towards longer-horizon temporality understanding, a stream free of appearance is necessary. Detailed comparison will be discussed in Sec. 2.4.5.3.

2.3.3 Action-centric Contrastive Learning (AcCL)

As aforementioned, question-answer contains much information including characters, objects, and locations. Actions, the important reasoning objective in videos, may only occupy a small portion of QA text and be neglected in the cross-modal alignment. Since VideoQA takes the alignment of global video features and a full QA sequence features as the optimization objective, the precise motion information obtained from Sec. 2.3.2 may not be well exploited. A VideoQA model, capable of answering temporal questions, should make good use of motion.

To this end, we propose a novel training scheme that conducts contrastive learning for visual-language matching before finetuning VideoQA objective. Different from conventional VL contrastive learning, the contrastive learning in our method is action-centric. It encourages the video representation to be aligned with the representation of **action phrase** that is parsed from the question. That is to say, other information such as entity, location, objects are not present in the text for matching. For example, in the question “what happens to the train after moving for a while?”, the action phrase to be aligned with the whole video clip is “moving for a while”. Under this matching objective, the video representation has to focus on precise motions, leading to a deep understanding of temporality. In specific, we propose a contrastive loss $\mathcal{L}_{pt}$ to update the encoders $\mathbb{F}_v, \mathbb{F}_q$:

$$\mathcal{L}_{pt} = \sum_i \log \left(\frac{\exp\left(s_c\right)}{\exp\left(s_c\right) + \sum_{c' \in \mathcal{N}_i} \exp\left(s'_c\right)} \right), \tag{2.5}$$

where $\mathcal{N}_i$ denotes the negative pool of action phrase for the i-th sample, *i.e.,* , action phrases from the questions that are unpaired to the video $\mathbf{h}$. $s_c = \mathcal{F}_v(\mathbf{h})\mathcal{F}_q(c)^\top$ is the similarity between the action phrase c and video $\mathbf{h}$ of the i-th sample. It encourages the video representation closer to its paired action phrase c and far away from the unpaired c' that are randomly sampled into the mini-batch. Thus, by contrastive to many other action phrases $c' \in \mathbb{N}_i$ in the dataset, the motion in vision and the textual action are better mined and aligned. The motion-plentiful features and model provide a good starting point for VideoQA finetuning.

Many VideoQA task benefits from contrastive learning based video language pretraining Lei et al. (2021); Zellers et al. (2021) from large-scale video-language data Bain et al. (2021), which is also reflected in the SoTA model in our task Xiao et al. (2022b). However, our AcCL is just conducted on our task datasets themselves, without resorting to any of the external training data, and has already been more effective than VGT Xiao et al. (2022b) with external data pretraining, while taking much less training resources.

2.3.4 Temporal Sensitivity-aware Confusion Loss

At the end of Sec. 2.3.2, we mention that although we have an appearance-free stream to extract precise motions, the appearance stream is indispensable. Unfortunately, the appearance stream, even fused with an appearance-free stream, provides the possibility to model action biased towards scene/object context Choi et al. (2019). To mitigate this issue, we propose to prevent the model from answering a question if the corresponding video is randomly ordered in the temporal domain. Our motivation is that the temporality reasoning needs the model to infer the inter-action relations across temporal, such as “stop (action 1) **after** moving for a while (action 2)”. Thus, if we randomly shuffle the video, the “after” relation no longer exists. A reliable network should be unable to answer the “stop” to the question like “What is the train doing after moving for a while?”.

Motivated by this, we design a confusion loss that takes as input the shuffled video $\tilde{\mathbf{h}}$ and question-answer $[q;a]$:

$$\begin{aligned}\mathcal{L}_{cf}^{(n)}(\hat{\mathbf{p}}, \hat{\mathbf{p}}) &= -\sum_{a=1}^{|\mathcal{A}|} \hat{p}^{(a)} \log \hat{p}^{(a)}, \\ \hat{p}^{(a)} &= \frac{\exp\left(\hat{s}^{(a)}\right)}{\sum_{k=1}^{|\mathcal{A}|} \exp\left(\hat{s}^{(k)}\right)},\end{aligned} \tag{2.6}$$

where $\hat{s}^{(a)} = \tilde{\mathbf{f}}_v \mathbf{f}_q^\top$ (denote the $\hat{s}^{(a)} \in [\hat{s}^{(1)}, \ldots, \hat{s}^{(|\mathcal{A}|)}]^\top$) is the inner-product similarity score for the a-th answer features $\mathbf{f}_q = \mathcal{F}_q\left([q;a]\right)$ *w.r.t.* its shuffled video feature vector $\tilde{\mathbf{f}}_v = \mathcal{F}_v(\tilde{\mathbf{h}})$. The confusion loss is applied on the shuffled video and encourages the largest entropy for all of answers, so that the scene context invariant to temporal order change will be ignored in action relation modeling.

Many questions in the VideoQAs, *e.g.,* “Where is the video taken?” just rely on descriptive content and can be answered even with the shuffled videos. Thus, the confusion loss only applies to temporal-sensitive questions, *e.g.,* the “after” question: ’what does A do after raising her hand?’ The temporal-sensitive questions contain specific English syntax, e.g. “before”, “after”, “when”. We filter out the temporal-insensitive questions based on the existence of the syntaxes. The overall optimization objective is as follows.

$$\min \mathbb{E}_{q^{(n)} \sim Q_\tau} \left[\mathcal{L}_{ce}^{(n)}(\mathbf{y}, \mathbf{p}) - \mathcal{L}_{cf}^{(n)}(\hat{\mathbf{p}}, \hat{\mathbf{p}}) \right], \tag{2.7}$$

where Q_τ denotes the set of questions that are temporally sensitive. $\mathcal{L}_{ce}^{(n)}$ is the cross entropy loss to metric if the probability over the candidates answers is $\mathbf{p} = [p^{(1)}, ..., p^{(|\mathcal{A}|)}]$ follows ground-truth answer y. $\mathcal{L}_{ce}^{(n)}$ is applied to all of the samples including the temporal-insensitive one, which is to optimize:

$$\min \mathbb{E}_{q^{(n)} \sim Q_{\setminus\tau}} \left[\mathcal{L}_{ce}^{(n)}(\mathbf{y}, \mathbf{p}) \right] \tag{2.8}$$

where $Q_{\setminus\tau}$ denotes the set of remaining temporally insensitive samples. The two loss are used for fine-tune the VideoQA after AcCL (see Sec. 2.3.3).

Methods	NExT-QA Val				NExT-QA Test			
	Acc@C	Acc@T	Acc@D	Acc@All	Acc@C	Acc@T	Acc@D	Acc@All
EVQA Antol et al. (2015)	42.46	46.34	45.82	44.24	43.27	46.93	45.62	44.92
STVQA Jang et al. (2017)	44.76	49.26	55.86	47.94	45.51	47.57	54.59	47.64
CoMem Gao et al. (2018)	45.22	49.07	55.34	48.04	45.85	50.02	54.38	48.54
HCRN* Le et al. (2020)	45.91	49.26	53.67	48.20	47.07	49.27	54.02	48.89
HME Fan et al. (2019)	46.18	48.20	58.30	48.72	46.76	48.89	57.37	49.16
HGA Jiang and Han (2020)	46.26	50.74	59.33	49.74	48.13	49.08	57.79	50.01
HQGA Xiao et al. (2022a)	48.48	51.24	61.65	51.42	49.04	52.28	59.43	51.75
P3D-G Cherian et al. (2022)	51.33	52.30	62.58	53.40	-	-	-	-
IGV Li et al. (2022c)	-	-	-	-	48.56	51.67	59.64	51.34
EIGV Li et al. (2022b)	-	-	-	-	-	-	-	53.70
ATPBuch et al. (2022)	53.1	50.2	66.8	54.30	-	-	-	-
VGT Xiao et al. (2022b)	52.28	55.09	64.09	55.02	51.62	51.94	63.65	53.68
VGT* Xiao et al. (2022b)	53.43	56.39	69.50	56.89	52.78	54.54	67.26	55.70
Ours	**56.04**	**58.44**	**65.38**	**58.27**	**55.31**	**55.55**	**65.34**	**57.03**

Table 2.1 **Results of multi-choice QA on validation set and test set of NextQA dataset.** The best results are bolded. Note that the greyed out VGT* uses 0.18 million videos from webvid dataset Bain et al. (2021) as pretraining, while the remaining include ATM do not pretrain on the external large-scale data. All of numbers for existing work are recorded from their papers. "-" indicates the missing results. Acc_C, Acc_T, Acc_D denote the accuracy for causality, temporality and descriptive questions.

2.4 Experiments

2.4.1 Datasets

NExT-QA Xiao et al. (2021) consists of 47.7K questions with answers in the form of multiple choices, which are annotated from 5.4K videos. It pinpoints the causal and temporal reasoning over the object interaction. Next-QA focuses on question answering with visual evidence. Thus, in addition to temporal reasoning questions, the causal questions *e.g.,* "How", "Why", require the corresponding answers visible in the video and also assess the multi-frame temporality event understanding.

TGIF-QA Jang et al. (2017) contains 134.7K questions about repeated actions, state transitions and a certain frame, which is annotated from 91.8K GIFs. **MSRVTT-QA** Xu et al. (2017) challenges a holistic visual recognition or description, which includes 10K annotated videos and 244K open-ended question-answer pairs.

2.4.2 Implementation Details

Appearance Features Following Xiao et al. (2022b,a), we decode the video into frames and sparsely sample 16 clips where each clip is in the length of 4 frames. To make a fair comparison with state-of-the-art VGT Xiao et al. (2022b), we also the RoI aligned features as the object appearance features $f_o \in \mathbb{R}^{16*2048}$, which is pretrained by Anderson et al. (2018). Frame features $f_i \in \mathbb{R}^{16*2048}$ are extracted by ResNet-50 He et al. (2016) pretrained on ImageNet.

Motion Features We use denseflow Wang et al. (2020) to extract the optical flow maps using videos' original FPS. Then, we use mmaction2 Contributors (2020)-based ResNet from TSN Wang et al. (2016) pre-trained on Kinetics-400 Carreira and Zisserman (2017) to extract optical flow features for the three datasets. To temporally align with the object and frame features, we uniformly distributed the flow maps into $K = 16$ clips per video. We uniformly sample 5 frames as per each clip and obtain a 2048-d feature vector for a clip. Thus, motion features f_m for a video are $\mathbb{R}^{16*2048}$.

Action-centric Contrastive Learning We parse the action phrases from questions using SpaCy parser Honnibal and Montani (2017). We use Adam optimizer Kingma and Ba (2015) with cosine annealing learning schedule of PyTorch initialized at $1e - 5$ on NVIDIA RTX A6000 at the maximum epoch of 10 among all of the datasets. Each batch contains 64 aligned video-action pairs and forms 64 pairs in total in the contrastive learning.

Fine-tuning We finetune the VideoQA using Adam optimizer Kingma and Ba (2015), batch size of 64, cosine annealing learning schedule of PyTorch initialized at $1e - 5$ on NVIDIA RTX A6000. The maximum epochs are set as 15 on NextQA, 30 on MSRVTT-QA and 50 on TGIFs.

2.4.3 Comparison with State-of-the-Art

Table 2.1 compares our method with existing state-of-the-art (SoTA) VideoQA methods on the widely used Next-QA dataset that feature the temporality reasoning. To ensure a fair comparison, ATM follows SoTA VGT Xiao et al. (2022b) and uses the exact same appearance feature extraction and applies DGT Xiao et al. (2022b) to model the object features. From the table, we can observe that ATM outperforms all existing methods without external data pretraining, by at least 3.85% and 3.35% on val. and test splits respectively. The outperformance is across causal, temporal, and descriptive splits of the Next-QA dataset, which demonstrate that ATM is effective in various question types that span from short segment to full video, from causal to temporal, and from single to repeated action execution.

Models	TGIF-QA				MSRVTT-QA
	Action	Transition	Action†	Transition†	
LGCN Huang et al. (2020)	74.3	81.1	-	-	-
HGA Jiang and Han (2020)	75.4	81.0	-	-	35.5
HCRN Le et al. (2020)	75.0	81.4	55.7	63.9	35.6
B2A Park et al. (2021)	75.9	82.6	-	-	36.9
HOSTR Dang et al. (2021)	75.0	83.0	-	-	35.9
HAIR Liu et al. (2021)	77.8	82.3	-	-	36.9
MASN Seo et al. (2021)	84.4	87.4	-	-	35.2
PGAT Peng et al. (2021)	80.6	85.7	58.7	65.9	38.1
MHN Peng et al. (2022)	83.5	90.8	-	-	38.6
ClipBERT* Lei et al. (2021)	82.8	87.8	-	-	37.4
SiaSRea* Yu et al. (2021)	79.7	85.3	-	-	41.6
MERLOT* Zellers et al. (2021)	94.0	96.2	-	-	**43.1**
VGT Xiao et al. (2022b)	95.0	**97.6**	59.9	70.5	39.7
Ours Zellers et al. (2021)	**96.0**	97.3	**65.7**	**71.0**	40.3

Table 2.2 **Results on TGIF-QA and MSVTT-QA.** † denotes TGIF-QA-R Peng et al. (2021) whose multiple choices for repeated action and state transition are more challenging. * denotes the models pretrained with large-scale external data.

Moreover, ATM which comes without external large-scale pre-training, even surpasses the existing method that used large-scale pretraining on more than 0.18 million videos Bain et al. (2021), by a clear margin of 1.38% and 1.33% on validation and test splits respectively. This demonstrates that ATM comprises of appearance-free motion features Sec 2.3.2, action-centric contrastive learning 2.3.3 and temporal sensitive-aware confusion objective 2.3.4, which holistically models action temporality, is more effective than the global video-text matching while uses less training computation resources.

In ATP Buch et al. (2022), the temporal modeling is performed on frames that are representative for single events and are encoded with CLIP model Radford et al. (2021). Our method also exceeds ATP Buch et al. (2022) by a large margin of 3.97%. This shows in temporality-heavy tasks, precise and faithful motion modeling is more effective than selecting the informative single frame for an event. This validates that ATM to precisely model and reason about motion, sets the new SoTA on Next-QA Xiao et al. (2021) benchmark.

We further compare ATM with SoTA on TGIF-QA in Table 2.2. Following the protocol, we use the same appearance features extracted by VGT Xiao et al. (2022b) and extract the motion stream features. We observe that ATM set new SoTA for repeated actions, and transition in TGIF-QA, which shows ATM as a whole is also effective in the repeated action and object transition scenarios.

For MSRVTT-QA in Table 2.2, our performance (free-of pertaining) is better than pretraining-free SoTA VGT but is inferior to the large-scale pre-trained methods MERLOT Zellers et al. (2021) and SiaSRea* Yu et al. (2021). This is because pre-training help model the descriptive content, while our work focuses on action temporality modeling.

2.4.4 True Temporality Metric

ATP Buch et al. (2022) evaluated the upper bound performance of a single-frame model on a video dataset and pointed out that even though NextQA dataset focuses on temporality reasoning, the dataset still contains static appearance bias. A small portion of questions can be correctly answered exclusively from a single frame without temporal information. To this end, we propose to measure the temporality faithfulness of VideoQA methods, *i.e.,* revealing if a VideoQA method learns true temporality to answering questions, instead of learning the correlation between the static appearance and the answer. In specific, the proposed true temporality metric measures the difference of QA accuracy between given the full video and given the middle frame respectively, as δ.

Table 2.3 shows that ATM better learns the true temporality compared to SoTA VGT, w/ w/o pretraining, on both Next-QA and TGif-QA. We observe that the external large-scale data for pretraining VGT guides the model to leverage more static information in temporality reasoning (only +0.84% on Next-QA test) since the pre-training helps more on the descriptive content that is static. Each of our component *i.e.,* AcCL, TSC, and appearance-free motion stream, helps to learn the true temporality. TSC mitigates the static bias by preventing answering temporality question if the temporal relations are destroyed. AcCL encourages learning motion representation agonistic to the entity or other appearance information. Appearance-free motion streams extract motion-plentiful representations that are necessary to understand the true temporality.

2.4.5 Ablation Studies

In addition to the study of each component, we conduct further ablation studies on NextQA Xiao et al. (2021) dataset.

	Next-QA (%)				TGIF-QA (%)			
	val-Acc	val-δ	test-Acc	test-δ	act-Acc	act-δ	trans-Acc	trans-δ
Ours	**58.27**	**+5.51**	**57.03**	**+5.13**	**96.0**	**+1.2**	97.3	**+1.3**
w/o AcCL	56.87	+2.71	55.02	+2.30	93.5	+0.5	97.1	+0.7
w/o TSC	57.99	+2.98	56.24	+3.25	95.6	+0.8	96.9	+0.2
w/o motion stream	56.57	+3.02	55.78	+2.80	95.2	+0.9	96.7	+0.8
VGT w/o pretrained	55.02	+2.91	53.68	+2.15	95.0	+0.6	**97.6**	+0.3
VGT w/ pretrained	56.89	+1.02	55.70	+0.84	-	-	-	-

Table 2.3 **True temporality evaluation**: Study of model components and comparison with SoTA.

2.4.5.1 Impact of Action-centric Contrastive Learning

We conduct an experiment where we test different variants of the text in Action-centric Contrastive Learning (AcCL). Table 2.4-a summarizes the results of the ablations. AcCL aims at learning action features by aligning the video with the action phrase from the question. The variants replace the action phrase by (1) the correct answer text *w.r.t.* to video-question, denoted as “Answer”, (2) the concatenation of the entire question and the correct answer text, denoted as “Question+Answer”, (3) the entire question text, denoted as “Question”, (4) the verb in question.

Variants	val (%)	test (%)
Action phrase (ours)	**58.27**	**57.03**
Answer	55.92	53.60
Question+Answer	56.83	55.16
Question	57.51	56.38
Verb in Question	57.07	56.57
w/o AcCL	56.87	55.02

(a)

Variants	test (%)	test-δ (%)
TS-aware (Ours)	**57.03**	**+5.13**
TS-unaware	56.89	+3.67
w/o TSC	56.24	+3.25

(b)

Variants	val (%)	test (%)
TSN (ours)	**58.27**	**57.03**
I3D	**57.71**	**56.40**
3D ResNext101	57.01	55.30
SlowFast	56.97	55.83
X3D	56.27	55.78
Timesformer	56.99	56.00
XCLIP	56.08	55.90
I3D-RGB only	57.35	55.63
TSN-RGB only	56.94	55.42
w/o motion stream	56.57	55.78

(c)

Table 2.4 Ablation study comparing different variants of (a) AcCL (b) TSC and (c) motion representations.

Table 2.4-a shows that our implementation of AcCL outperforms all of the other variants. We observe that the “Question” variant performs 0.65% worse than our “action in question" on test split since the full question text contains entity, scene, and other appearance information in addition to the action phrase. Contrasting with full questions will distract the representation from the motion information to the dominant and easily learned appearance features, which is less effective than action-centric version. Using “Answer”, “Question+Answer” also performs worse than ours. This demonstrates that the action phrases in questions are the information that the randomly initialized model parameters easily overlook but are important for temporality. Using “verb from question” is also less effective, as the action cannot be described by a single word in many cases, e.g. verb “get” is not informative enough for the action “get up”.

2.4.5.2 Impact of TSC Loss

We compare our Temporal Sensitivity-aware Confusion loss (TSC) in Table 2.4-b, with variants (1) removing the TSC and only training with cross-entropy, as “w/o TSC”. (2) applying the confusion loss to all samples regardless of time-sensitivity, as “TS-unaware”. Our method is slightly better than these two variants in VideoQA accuracy and much higher on the proposed true temporality reasoning metric. This validates that alleviating static bias by TSC helps a faithful temporal reasoning model, which in turn improves the event temporality understanding.

2.4.5.3 Impact of Appearance-free stream

Table 2.4-c shows the ablations on motion features f_m and analyzes the effectiveness of incorporating an appearance-free stream. In the table, TSN and I3D extract motion features with an appearance-free stream *i.e.,* flow maps, while the remaining extract motions only from the appearance-included input *i.e.,* RGB. These RGB-only methods SlowFast Feichtenhofer et al. (2019), X3D Feichtenhofer (2020), TimeSformer Bertasius et al. (2021) and XCLIP Ni et al. (2022) show superb performance on action recognition, as shown in Fig. 2.3-b. But they fall behind of the methods with the optical flow on motion feature extraction for VideoQA, though TSN and I3D are relatively early work without fancy network structures. RGB frames may only be enough for characterizing limited sets of atomic actions that are dominant for action recognition, it is less effective in modeling events with long horizon temporality. 3D ResNext101 Hara et al. (2017) has been used for motion feature extraction in existing VideoQA Le et al. (2020); Xiao et al. (2022a), but it is also RGB-only and 1.73% worse than TSN where flow is used.

2.4.6 Qualitative Analysis

In Fig. 2.4, we qualitatively assess the effect of the ATM by visualizing the results of representative samples in val. split. We can observe that the AcCL scheme helps to learn the discriminative

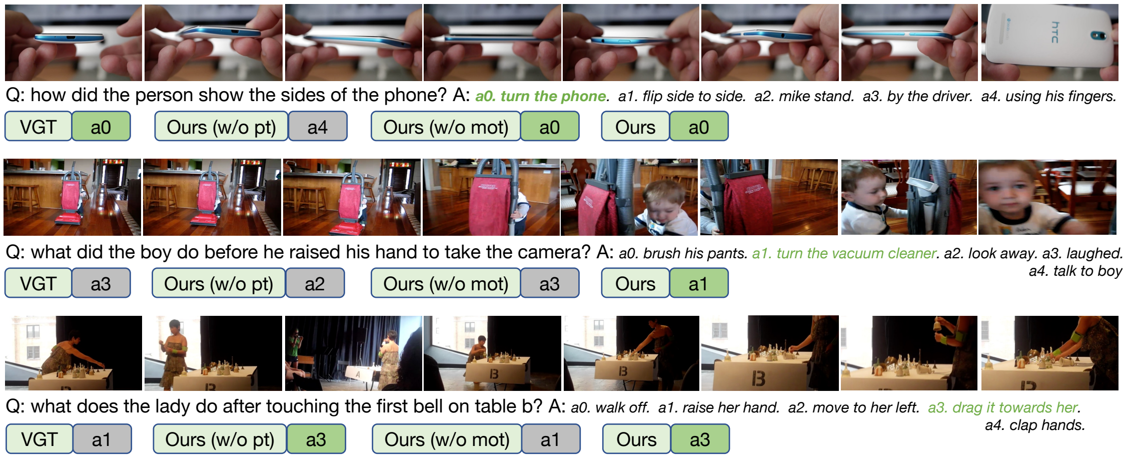

Q: how did the person show the sides of the phone? A: a0. turn the phone. a1. flip side to side. a2. mike stand. a3. by the driver. a4. using his fingers.
VGT
a0
Ours (w/o pt)
a4
Ours (w/o mot)
a0
Ours
a0
Q: what did the boy do before he raised his hand to take the camera? A: a0. brush his pants. a1. turn the vacuum cleaner. a2. look away. a3. laughed. a4. talk to boy
VGT
a3
Ours (w/o pt)
a2
Ours (w/o mot)
a3
Ours
a1
Q: what does the lady do after touching the first bell on table b? A: a0. walk off. a1. raise her hand. a2. move to her left. a3. drag it towards her. a4. clap hands.
VGT
a1
Ours (w/o pt)
a3
Ours (w/o mot)
a1
Ours
a3

Figure 2.4 Visualization. The ground-truth are marked in green. We display the results of ATM, ATM w/o AcCL (as "Ours w/o AcCL"), ATM w/o motion stream (as "Ours w/o flow") and the existing SoTA method VGT. The samples span across causality (1) and temporality (2, 3) reasoning.

representations for actions *e.g.,* "turn" in (1), while the variant w/o AcCL may learn the superficial correlations between appearance *e.g.,* "his fingers" and the answers. Moreover, the appearance-free stream also helps in extracting precise and useful motions. Since the scene and actor do not change in (3), the optical flow stream is informative for recognizing the "drag towards" action.

2.5 Summary

In this paper, we propose a novel framework to solve the VideoQA. Our method addresses the importance of temporality reasoning. To this end, we realize that it is worth revisiting optical flow, as flow may become less considered in atomic action recognition but is still effective in long-horizon temporality. Then, we propose an action-centric contrastive learning that makes both video and text representations informative for action. Then, we fine-tune the VideoQA via a novel temporal sensitivity-aware confusion loss to mitigate the potential static bias. Our ATM method is demonstrated to be superior to all existing VideoQA methods on multiple benchmarks and shows a faithful temporality reasoning via a new metric.

BIBLIOGRAPHY

Anderson, P., He, X., Buehler, C., Teney, D., Johnson, M., Gould, S., and Zhang, L. (2018). Bottom-up and top-down attention for image captioning and visual question answering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6077–6086.

Antol, S., Agrawal, A., Lu, J., Mitchell, M., Batra, D., Lawrence Zitnick, C., and Parikh, D. (2015). Vqa: Visual question answering. In *CVPR*, pages 2425–2433.

Bain, M., Nagrani, A., Varol, G., and Zisserman, A. (2021). Frozen in time: A joint video and image encoder for end-to-end retrieval. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1728–1738.

Bertasius, G., Wang, H., and Torresani, L. (2021). Is space-time attention all you need for video understanding?

Buch, S., Eyzaguirre, C., Gaidon, A., Wu, J., Fei-Fei, L., and Niebles, J. C. (2022). Revisiting the" video" in video-language understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2917–2927.

Carreira, J. and Zisserman, A. (2017). Quo vadis, action recognition? A new model and the Kinetics dataset. In *CVPR*.

Cherian, A., Hori, C., Marks, T. K., and Le Roux, J. (2022). (2.5+ 1) d spatio-temporal scene graphs for video question answering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 444–453.

Choi, J., Gao, C., Messou, J. C., and Huang, J.-B. (2019). Why can't i dance in the mall? learning to mitigate scene bias in action recognition. 32.

Contributors, M. (2020). Openmmlab's next generation video understanding toolbox and benchmark. https://github.com/open-mmlab/mmaction2.

Dang, L. H., Le, T. M., Le, V., and Tran, T. (2021). Hierarchical object-oriented spatio-temporal reasoning for video question answering.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding.

Ding, G. and Yao, A. (2021). Temporal action segmentation with high-level complex activity labels. *arXiv preprint arXiv:2108.06706*.

Duan, H., Zhao, Y., Chen, K., Lin, D., and Dai, B. (2022). Revisiting skeleton-based action recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern*

Recognition, pages 2969–2978.

Fan, C., Zhang, X., Zhang, S., Wang, W., Zhang, C., and Huang, H. (2019). Heterogeneous memory enhanced multimodal attention model for video question answering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1999–2007.

Feichtenhofer, C. (2020). X3d: Expanding architectures for efficient video recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 203–213.

Feichtenhofer, C., Fan, H., Malik, J., and He, K. (2019). Slowfast networks for video recognition. In *ICCV*.

Gao, J., Ge, R., Chen, K., and Nevatia, R. (2018). Motion-appearance co-memory networks for video question answering. In *CVPR*, pages 6576–6585.

Hara, K., Kataoka, H., and Satoh, Y. (2017). Can spatiotemporal 3d cnns retrace the history of 2d cnns and imagenet? *arXiv:1711.09577*.

He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778.

Honnibal, M. and Montani, I. (2017). spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing. To appear.

Huang, D., Chen, P., Zeng, R., Du, Q., Tan, M., and Gan, C. (2020). Location-aware graph convolutional networks for video question answering. In *AAAI*, pages 11021–11028.

Jang, Y., Song, Y., Yu, Y., Kim, Y., and Kim, G. (2017). Tgif-qa: Toward spatio-temporal reasoning in visual question answering. In *CVPR*.

Jiang, P. and Han, Y. (2020). Reasoning with heterogeneous graph alignment for video question answering. In *AAAI Conference on Artificial Intelligence (AAAI)*.

Kay, W., Carreira, J., Simonyan, K., Zhang, B., Hillier, C., Vijayanarasimhan, S., Viola, F., Green, T., Back, T., Natsev, P., et al. (2017). The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*.

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

Le, T. M., Le, V., Venkatesh, S., and Tran, T. (2020). Hierarchical conditional relation networks for video question answering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9972–9981.

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

Lei, J., Li, L., Zhou, L., Gan, Z., Berg, T. L., Bansal, M., and Liu, J. (2021). Less is more: Clipbert for video-and-language learning via sparse sampling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7331–7341.

Lei, J., Yu, L., Bansal, M., and Berg, T. L. (2018). Tvqa: Localized, compositional video question answering.

Lei, J., Yu, L., Berg, T. L., and Bansal, M. (2019). Tvqa+: Spatio-temporal grounding for video question answering.

Li, J., Niu, L., and Zhang, L. (2022a). From representation to reasoning: Towards both evidence and commonsense reasoning for video question-answering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21273–21282.

Li, L., Chen, Y.-C., Cheng, Y., Gan, Z., Yu, L., and Liu, J. (2020). Hero: Hierarchical encoder for video+ language omni-representation pre-training. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2046–2065.

Li, Y., Li, Y., and Vasconcelos, N. (2018). Resound: Towards action recognition without representation bias. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 513–528.

Li, Y., Song, Y., Cao, L., Tetreault, J., Goldberg, L., Jaimes, A., and Luo, J. (2016). Tgif: A new dataset and benchmark on animated gif description. In *CVPR*, pages 4641–4650.

Li, Y., Wang, X., Xiao, J., and Chua, T.-S. (2022b). Equivariant and invariant grounding for video question answering. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 4714–4722.

Li, Y., Wang, X., Xiao, J., Ji, W., and Chua, T.-S. (2022c). Invariant grounding for video question answering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2928–2937.

Lin, J., Gan, C., and Han, S. (2019). Tsm: Temporal shift module for efficient video understanding. In *ICCV*, pages 7083–7093.

Liu, F., Liu, J., Wang, W., and Lu, H. (2021). Hair: Hierarchical visual-semantic relational reasoning for video question answering. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1698–1707.

Liu, Y., Albanie, S., Nagrani, A., and Zisserman, A. (2019). Use what you have: Video retrieval using representations from collaborative experts. *arXiv preprint arXiv:1907.13487*.

Luo, H., Ji, L., Zhong, M., Chen, Y., Lei, W., Duan, N., and Li, T. (2021). Clip4clip: An empirical study of clip for end to end video clip retrieval. In *arXiv preprint arXiv:2104.08860*.

Ni, B., Peng, H., Chen, M., Zhang, S., Meng, G., Fu, J., Xiang, S., and Ling, H. (2022). Expanding language-image pretrained models for general video recognition. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part IV*, pages 1–18. Springer.

Park, J., Lee, J., and Sohn, K. (2021). Bridge to answer: Structure-aware graph interaction network for video question answering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 15526–15535.

Peng, L., Yang, S., Bin, Y., and Wang, G. (2021). Progressive graph attention network for video question answering. In *ACM MM*, pages 2871–2879.

Peng, M., Wang, C., Gao, Y., Shi, Y., and Zhou, X.-D. (2022). Multilevel hierarchical network with multiscale sampling for video question answering. *IJCAI*.

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

Seo, A., Kang, G.-C., Park, J., and Zhang, B.-T. (2021). Attend what you need: Motion-appearance synergistic networks for video question answering. In *ACL*, pages 6167–6177.

Shao, D., Zhao, Y., Dai, B., and Lin, D. (2020). Finegym: A hierarchical video dataset for fine-grained action understanding. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2616–2625.

Soomro, K., Zamir, A. R., and Shah, M. (2012). Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*.

Wang, L., Xiong, Y., Wang, Z., Qiao, Y., Lin, D., Tang, X., and Van Gool, L. (2016). Temporal segment networks: Towards good practices for deep action recognition. In *ECCV*.

Wang, S., Li, Z., Zhao, Y., Xiong, Y., Wang, L., and Lin, D. (2020). denseflow. https://github.com/open-mmlab/denseflow.

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

Xiao, J., Yao, A., Liu, Z., Li, Y., Ji, W., and Chua, T.-S. (2022a). Video as conditional graph hierarchy for multi-granular question answering. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, pages 2804–2812.

Xiao, J., Zhou, P., Chua, T.-S., and Yan, S. (2022b). Video graph transformer for video question answering. In *European Conference on Computer Vision*, pages 39–58. Springer.

Xu, D., Zhao, Z., Xiao, J., Wu, F., Zhang, H., He, X., and Zhuang, Y. (2017). Video question answering via gradually refined attention over appearance and motion. In *ACM MM*, pages 1645–1653.

Xu, J., Mei, T., Yao, T., and Rui, Y. (2016). Msr-vtt: A large video description dataset for bridging video and language. In *CVPR*.

Yang, A., Miech, A., Sivic, J., Laptev, I., and Schmid, C. (2021a). Just ask: Learning to answer questions from millions of narrated videos. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1686–1697.

Yang, A., Miech, A., Sivic, J., Laptev, I., and Schmid, C. (2021b). Just ask: Learning to answer questions from millions of narrated videos. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1686–1697.

Yu, W., Zheng, H., Li, M., Ji, L., Wu, L., Xiao, N., and Duan, N. (2021). Learning from inside: Self-driven siamese sampling and reasoning for video question answering. 34.

Yu, Z., Xu, D., Yu, J., Yu, T., Zhao, Z., Zhuang, Y., and Tao, D. (2019). Activitynet-qa: A dataset for understanding complex web videos via question answering.

Zellers, R., Lu, X., Hessel, J., Yu, Y., Park, J. S., Cao, J., Farhadi, A., and Choi, Y. (2021). Merlot: Multimodal neural script knowledge models. In *Advances in neural information processing systems (NeurIPS)*, volume 34.

Zhou, H., Kadav, A., Shamsian, A., Geng, S., Lai, F., Zhao, L., Liu, T., Kapadia, M., and Graf, H. P. (2021). Composer: Compositional reasoning of group activity in videos with keypoint-only modality. *arXiv preprint arXiv:2112.05892*.

Zhu, L. and Yang, Y. (2020). Actbert: Learning global-local video-text representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8746–8755.

CHAPTER 3

GROUP ACTIVITY PREDICTION WITH SEQUENTIALLY RELATIONAL ANTICIPATION MODEL

3.1 Background and Motivation

Group activity prediction is to forecast an activity performed by a group of people before the activity ends. Different from group activity recognition, it only has access to the beginning frames of a video containing **incomplete** activity execution. It is useful in scenarios where the intelligent systems have to make prompt decisions, such as surveillance and traffic accident avoidance where multiple people are present. Unfortunately, existing action prediction methods Yao et al. (2018); Yan et al. (2017); Kong et al. (2017, 2014, 2018) are limited to actions performed by an individual. Even though some methods Kong et al. (2017, 2020); Wang et al. (2019) attempt to predict actions performed by multiple people in standard databases such as UCF101 Soomro et al. (2012), they simply model the multiple people as a single entity and ignore their relations. This would undoubtedly result in a low prediction performance.

As shown in Kong et al. (2017), one of the major challenges in activity prediction is how to enhance the discriminative power of the features extracted from the partial observations. However, this is even more challenging to do so in group activity prediction as multiple people are present in the scene. Each person's individual action may vary and people's interactions frequently appear and change in a group activity. To this end, it is important to model the relations of multiple people in the observed frames and predict their future group representations. In addition, if only limited beginning frames are observed, it would be extremely difficult to directly anticipate the features of full observations at once. A temporally progressive anticipation model is desired for modeling activity evolution.

To address these challenges, we propose a novel sequential relational anticipation model (SRAM) for group activity prediction by anticipating group activities and positions in the future (see Fig. 3.1). SRAM is developed as an encoder-decoder framework, in which an *observation encoder* summarizes the relational dynamics in these beginning observed frames and a *sequential* *decoder* further anticipates the representations for group activities and positions occurring in the future. Specifically, the observation encoder naturally models the relational dynamics of people and complex interactions between people in the observed frames. To predict group activity, we propose a sequential decoder to anticipate the structured group representation in the future using several unrolling stages. Two graph auto-encoders are used in the sequential decoder to explicitly anticipate the activity and the position relations of people in the unobserved frames. We propose to make a sequential prediction that *progressively* anticipates the future group representation by performing multiple unrolling stages guided by three novel loss functions. This allows us to better capture complex group activity evolution.

To our best knowledge, we are the first to investigate the challenging problem of group activity prediction. The benefit of our method is twofold. Firstly, it not only predicts people's group activities but also predicts individuals' positions in the future. Our experimental results show that predicting people's future positions significantly helps predict their group activities. Secondly, the proposed method progressively anticipates structured group representations, which has shown to be very powerful in prediction especially when limited frames are observed. This idea could be generalized to other prediction tasks, e.g. human motion prediction Martinez et al. (2017) and video prediction Wichers et al. (2018).

Our contributions can be summarized as follows:

- We propose a novel sequential decoder to anticipate the representations for multiple people's future positions and activity, aiming to learn a discriminative structural representation for group activity prediction.

- We progressively anticipate the structured group representations at several unrolling stages guided by novel loss functions. This improves the performance when only few frames are observed.

- Extensive experiments demonstrate that our method outperforms the existing state-of-the-arts by a large margin.

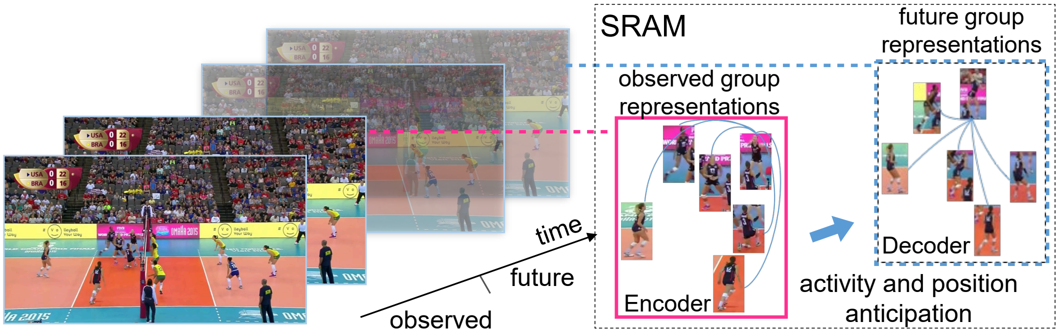

SRAM
future group representations
observed group representations
Decoder
Encoder
activity and position anticipation
time
future
observed

Figure 3.1 Given the beginning frames, our method models the relational dynamics of a group, and predicts a group activity by anticipating the group activity representation and their positions occurring in the future unobserved frames.

3.2 Related Work

Action Prediction aims to recognize the label of an action before the action is fully executed. Existing work Lan et al. (2014); Cai et al. (2019); Kong et al. (2018); Hu et al. (2018); Sadegh Aliakbarian et al. (2017); Ma et al. (2016); Zhao and Wildes (2019); Shi et al. (2018); Vondrick et al. (2016) focuses on predicting actions performed by an individual. Ryoo (2011) used integral and dynamic bag-of-words to represent features variations over time. DeepSCN Kong et al. (2017) and AAPNet Kong et al. (2020) make use of sequential context information by transferring knowledge in full videos to partial observations. Wang et al. (2019) developed a teacher-student learning framework to distill knowledge from the action recognition task, in order to enhance action prediction. Gammulle et al. (2019) presented a jointly learnt task for both action prediction and future motion representation inference.

Prediction on interactions between two people was studied in Yan et al. (2017); Yao et al. (2018). Yan et al. (2017) developed a tri-coupled recurrent structure and an attention mechanism to address action prediction for two individuals' interactions. Yao et al. (2018) predicted the motion of the interactions between two people, but did not predict their interaction labels. Different from them, we focus on the prediction of group activities involving multiple people. Our method elegantly captures complex relational dynamics between people for learning discriminative information.

Group Activity Recognition has been extensively studied in previous work Amer et al. (2014); Shu et al. (2017); Ibrahim et al. (2016); Yan et al. (2018a). Early work applies graphical models on the extracted hand-craft features Amer et al. (2014); Lan et al. (2012); Wang et al. (2013) as group representations. Deep learning methods for multi-people activity recognition have shown excellent performance Shu et al. (2017); Bagautdinov et al. (2017); Wang et al. (2017); Ibrahim and Mori (2018); Deng et al. (2016); Tang et al. (2018); Gavrilyuk et al. (2020). HDTM Ibrahim et al. (2016) develops a two-stage LSTM model to firstly extract features of temporal individual motions and then aggregate neighborhood information. SSU Bagautdinov et al. (2017) achieves the individual detection and group activity recognition in a unified framework. Recent work suggests that only part of people's motions contribute to the entire group activity Yan et al. (2018a); Gammulle et al. (2018); Ramanathan et al. (2016); Azar et al. (2019); Hu et al. (2020), via suppressing the irrelevant actions. Previous work also shows that interactions between people are important in understanding group activity. For example, HRNIbrahim and Mori (2018) introduces a hierarchical spatial relational layer to learn the relational representations between two players. Other methods, including Stagnet Qi et al. (2018), S-RNN Biswas and Gall (2018), SBGAR Li and Choo Chuah (2017) apply structural-RNN to obtain spatiotemporal features. ARG Wu et al. (2019) explicitly models the interactions by employing graph convolution on a learnable graph.

The main difference between our work and group activity recognition methods is that we aim at predicting the group activity label given *incomplete* activity execution, while these methods are given complete activity executions. This prompts us to develop novel model architecture and loss functions in this work.

3.3 Proposed Method – Sequential Relational Anticipation Model

Problem formulation. Our goal is to predict the activity label y of a group of people given a partial observation of a video containing incomplete activity execution. We define the *observation ratio* as the number of observed frames t_0 in a streaming video divided by the total number of frames T in the corresponding full video following Kong et al. (2017, 2020), *i.e.,* t_0/T. For instance, if a partial video contains 30 frames and the corresponding full video contains 100 frames, then the observation ratio of this activity is 30%.

During training, we have access to all full training videos containing complete group activity

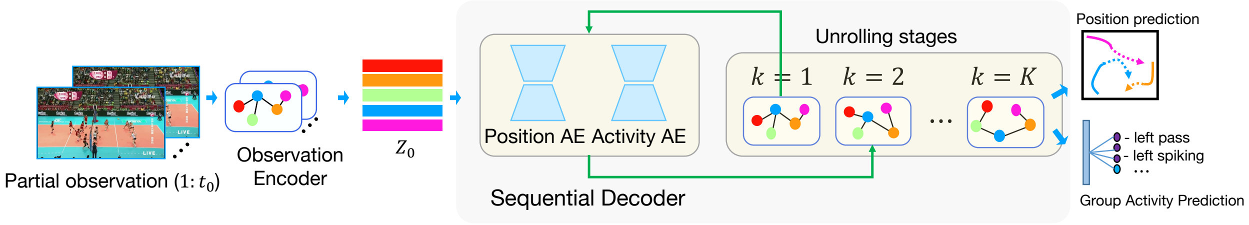


Figure 3.2 Overall architecture. Our framework SRAM takes the beginning t_0 observed frames as input and predicts the group activity label. An observation encoder first summarizes the relational dynamics in partial observation as a latent variable Z_0. Then, a sequential decoder takes over Z_0 and progressively anticipates the group representation through K unrolling stages. The output of the last unrolling stage is expected to contain rich discriminative information for group activity prediction. Details can be seen in Fig. 3.3.

executions. These full videos are supposed to contain all the discriminative information for classification. During test, given a partial observation of a group activity execution, we encourage our model to anticipate the group representations that contain similar amount of discriminative information as the corresponding full observation. Thus, its prediction power can be enhanced.

Overall architecture. The overall architecture is shown in Fig. 3.2. We formulate our group activity prediction model as an encoder-decoder framework that contains an *observation encoder* and a *sequential decoder*. Given a partial observation containing t_0 frames, the observation encoder summarizes the relational dynamics of the group from the partial observation and then the sequential decoder anticipates the group representation for activities and positions in the future unobserved frames.

Due to the large motion variations between a partial and a full observation, a novel sequential decoder is proposed in this work to progressively anticipate the structured group representation for the future unobserved frames by several unrolling stages. This is useful for enhancing the discriminative power of the anticipated representation, especially if limited frames are observed. Moreover, for group activity, relations between multiple people are discriminative information and they vary as time. To predict group activity, our sequential decoder uses two graph auto-encoders to concurrently perform relational anticipation on both people's activity features and their positions.

3.3.1 Relation Modeling for Group Activity

Given t_0 observed frames, we first extract features of all the observed t_0 frames, and then apply ROIAlign He et al. (2017) to extract the feature vectors of multiple people based on their positions $\{B_1, B_2, \cdots, B_{t_0}\}(t \in \{1, \cdots, t_0\})$. Action features and positions of the i-th individual on the t-th frame are represented as $\mathbf{x}_t(i)$ and $\mathbf{b}_t(i)$ respectively. Afterwards, upon the individual dynamics, we follow Wu et al. (2019) to explicitly model the pair-wise *position relations* and *action relations* of multiple people in the observed frames as two relation graphs $G_t^{\mathrm{a}} \in \mathbb{R}^{N\times N}$ and $G_t^{\mathrm{p}} \in \mathbb{R}^{N\times N}$, respectively. Both of the two graphs have N nodes representing N people in the t-th frame. Given the i-th and j-th individuals, the edge of the action similarity graph $G_t^{\mathrm{a}}(i, j)$ is computed by the cosine similarity and normalized by Softmax function. The edge on the position relation graph $G_t^{\mathrm{p}}(i, j)$ is computed by the normalized Euclidean distance (denoted by $d(\cdot, \cdot)$):

$$G_t^{\mathrm{a}}(i, j) = \frac{\exp\left(\mathbf{x}_t(i)^{\mathrm{T}} \cdot \mathbf{x}_t(j)\right)}{\sum_{j=1}^{N} \exp\left(\mathbf{x}_t(i)^{\mathrm{T}} \cdot \mathbf{x}_t(j)\right)},\ G_t^{\mathrm{p}}(i, j) = \frac{1/d(\mathbf{b}_t(i), \mathbf{b}_t(j))}{\sum_{j=1}^{N} 1/d(\mathbf{b}_t(i), \mathbf{b}_t(j))}. \tag{3.1}$$

Once the graphs are built, we obtain the structured representations for the group activity in the observed frames. We will also perform anticipation on the two graphs representing the group activity in the unobserved frames, which will be discussed below.

3.3.2 Observation Encoder $\mathcal{E}$

The observation encoder $\mathcal{E}$ is proposed to summarize spatiotemporal information of the complex relational dynamics of multiple people in partial observations containing t_0 frames. $\mathcal{E}$ learns to map $G_{1:t_0}^{\mathrm{a}}$, $G_{1:t_0}^{\mathrm{p}}$, and $X_{1:t_0}$ to a latent variable Z_0, by the spatio-temporal graph convolution network ST-GCN Yan et al. (2018b). Specifically, it first performs spatial graph convolution Kipf and Welling (2017) on the two graphs G_t^{p} and G_t^{a} for the t-th frame

$$\sigma(G_t^{\mathrm{p}} X_t W_{\mathrm{p}}) + \sigma(G_t^{\mathrm{a}} X_t W_{\mathrm{a}}), \tag{3.2}$$

and then performing temporal convolution Lea et al. (2017) on every three consecutive frames to learn the latent variable Z_0. Here, σ is ReLU activation, W_{p} and W_{a} are learnable weights, and X_t is the action features of N people. Latent variable Z_0 will be integrated in the sequential decoder, and guides its unrolling stages.

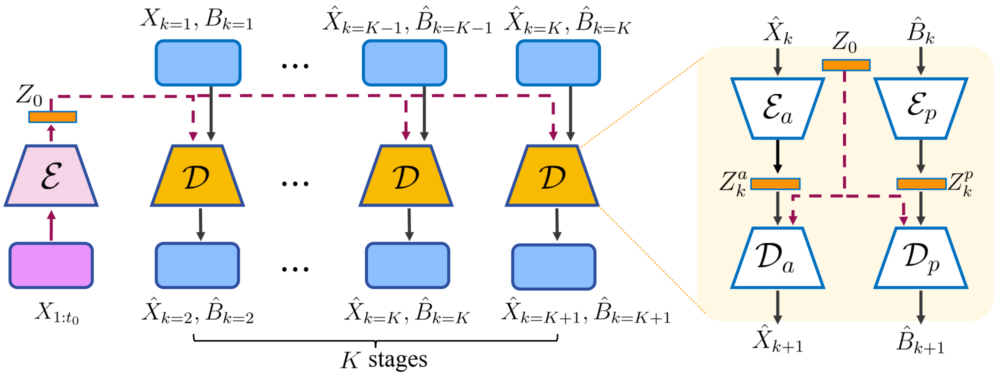


Figure 3.3 Sequential decoder $\mathcal{D}$ is formulated as two auto-encoders $\mathcal{E}_{\mathrm{a}}$-$\mathcal{D}_{\mathrm{a}}$ and $\mathcal{E}_{\mathrm{p}}$-$\mathcal{D}_{\mathrm{p}}$ that progressively anticipate the group activity representation for future unobserved frames using multiple unrolling stages. At the k-th stage, $\mathcal{D}$ is fed with the summary of the partial observation encoded in the latent variable Z_0 as well as the action features $\hat{X}_k$ and the position features $\hat{B}_k$ from the previous stage. Then $\mathcal{D}$ anticipates the action features $\hat{X}_{k+1}$ and positions $\hat{B}_{k+1}$.

Different from ARG Wu et al. (2019), our model captures the temporal patterns of people's relations, which is useful for group activity prediction.

3.3.3 Sequential Decoder $\mathcal{D}$

The performance of state-of-the-art action prediction methods Kong et al. (2020, 2017) is still limited especially when few beginning frames are given. This is mainly because they use direct mapping from partial observation to the corresponding full observation in one pass, which is not powerful enough to deal with large visual variations between partial and full observations. In this paper, we propose a sequential decoder that *progressively* anticipates the group representation that is expected to contain rich discriminative information as the fully observed activity using K unrolling stages (see Fig. 3.3). This allows us to create a more powerful model for group activity prediction.

Besides, different from individual action prediction methods Kong et al. (2020); Wang et al. (2019), people's relations formulated as graphs using Eq. (3.1), are discriminative information for group activity. Moreover, the group activity varies overtime. It is necessary to predict group representations by anticipating relations in the unobserved stage. As described in Chapter 3.3.1, people's relations can be inferred from their action similarity and relative positions. For example, a partial observation of a volleyball activity is given, which contains run-up of ace spikers and waiting gestures of their opponents. Our model is supposed to predict it as "spiking" by the cue that the players are moving towards net with their actions. Therefore, we develop a sequential decoder as a mixture of two graph auto-encoders: an activity auto-encoder $\mathcal{E}_{\mathrm{a}}$-$\mathcal{D}_{\mathrm{a}}$ for predicting activity representations and a position auto-encoder $\mathcal{E}_{\mathrm{p}}$-$\mathcal{D}_{\mathrm{p}}$ for predicting positions of multiple people. The two auto-encoders are coupled by the shared latent variables Z_0 learned from partial observations.

Activity auto-encoder $\mathcal{E}_{\mathbf{a}}$-$\mathcal{D}_{\mathbf{a}}$. Using K activity auto-encoders, the proposed sequential decoder progressively anticipates the activity representation by K unrolling stages. Each activity auto-encoder at the k-th stage ($k \in \{1, 2, \cdots, K\}$) is fed with the output $\hat{X}_k$ of the activity auto-encoder at the previous $(k-1)$-th stage. We use the spatiotemporal action features at the last observed frame t_0 as the input of the activity auto-encoder at stage $k = 1$. We encode the input $\hat{X}_k$ of current unrolling stage to a latent variable Z_k^{a} by

$$Z_k^{a} = \sigma(G_k^{\mathrm{p}} \hat{X}_k U_{\mathrm{ep}}) + \sigma(G_k^{\mathrm{a}} \hat{X}_k U_{\mathrm{ea}}), \tag{3.3}$$

and then decodes the activity representation $\hat{X}_{k+1}$:

$$\hat{X}_{k+1} = \sigma(G_k^{\mathrm{a}}(Z_0 + Z_k^{\mathrm{a}}))U_{\mathrm{da}}) + \sigma(G_k^{\mathrm{p}}(Z_0 + Z_k^{\mathrm{a}})U_{\mathrm{dp}}), \tag{3.4}$$

where U_{ep}, U_{ea}, U_{dp}, U_{da} are learnable parameters. $\hat{X}_{k+1}$ is the anticipated group features at the k-th stage, and is served as the input for the activity auto-encoder at the $(k+1)$-th stage. The anticipation of $\hat{X}_{k+1}$ is conditioned on latent variables Z_k^{a} and Z_0, in order to both keep track of the short-term information of the previous unrolling stage and use the long-term spatiotemporal information in the partial observations. G_k^{a} and G_k^{p} are computed by the generated activity features and positions at the k-th stage using similar functions as Eq. (3.1), respectively (replacing time step t by the stage k).

The benefits of the progressive anticipation using K unrolling stages lie in two aspects. First, the temporal dependency of activity evolution is naturally built between successive stages. This allows us to naturally anticipate structured group activity representations for prediction purpose. Second, the prediction granularity can be controlled with the number of unrolling stages K. The case when $K = 1$ is equivalent to the existing one-pass solution used in Kong et al. (2020, 2017).

Position auto-encoder $\mathcal{E}_\mathbf{p}$-$\mathcal{D}_\mathbf{p}$. As described in Chapter 3.3.1, the interactions between two people also depend on their relative positions. Thus, it is necessary to explicitly anticipate the positions of these people in group activity prediction.

Similar to activity auto-encoder, the proposed sequential decoder also performs K unrolling stages for position prediction for a group of people using K position auto-encoders. Each position auto-encoder at stage k is fed with the output of its previous auto-encoder at stage $k-1$, and outputs the positions of people. Experimental results in Chapter 3.4.4 show that the anticipated future positions of people help improve performance of group activity prediction.

The position auto-encoder first encodes the positions $\hat{B}_k$ of multiple people to a latent variable Z_k^{p} at stage k through graph convolution Kipf and Welling (2017):

$$Z_k^{\mathrm{p}} = \sigma(G_k^{\mathrm{p}} \hat{B}_k V_{\mathrm{ep}}) + \sigma(G_k^{\mathrm{a}} \hat{B}_k V_{\mathrm{ea}}), \tag{3.5}$$

and then decodes the positions $\hat{B}_{k+1}$ for the next stage by

$$\hat{B}_{k+1} = \sigma(G_k^{\mathrm{a}}(Z_0 + Z_k^{\mathrm{p}}))V_{\mathrm{da}}) + \sigma(G_k^{\mathrm{p}}(Z_0 + Z_k^{\mathrm{p}})V_{\mathrm{dp}}), \tag{3.6}$$

where V_{ep}, V_{ea}, V_{dp}, V_{da} are learnable parameters. G_k^{p} and G_k^{a} are the same graphs used in the activity auto-encoder. The anticipation of $\hat{B}_{k+1}$ is conditioned on latent variables Z_k^{p} and Z_0, in order to both keep track of the short-term position information of the previous unrolling stage and use the long-term spatiotemporal information in the partial observations.

Position prediction is also benefited by sequential prediction via several unrolling stages, since the prediction granularity can be controlled. Similar to the activity auto-encoder, the position auto-encoder at stage $k = 1$ also takes the positions B_{t_0} of people on the last observed frame as input. The activity auto-encoder and the position auto-encoder share the same graphs G_k^{p} and G_k^{a} and are both conditioned on the latent variable Z_0 (see Fig. 3.3).

3.3.4 Feature Aggregation for Prediction

SRAM returns both group activity and position representations at each of K unrolling stages. The K-th stage corresponds to the full observation status, which contains the most discriminative information of an activity. We disregard all the outputs given by the activity autoencoders from the 1-st to $(K-1)$-th stages, and perform max-pooling on the output $\hat{X}_{K+1}$ given by the activity autoencoder at the K-th stage as the group activity representations. The resulting feature vector is used for group activity prediction. Similarly, we directly use the output $\hat{B}_{K+1}$ given by the K-th position autoencoder to perform position prediction.

3.3.5 Loss Functions and Model Learning

Adversarial loss. Inspired by Goodfellow et al. (2014), we encourage SRAM to generate representations corresponding to ground-truth full observations. We use two discriminators for the features generated by the sequential decoder. Discriminator D_1 is an activity classifier implemented by a softmax layer. $\mathcal{L}_{\text{cls}}$ is computed on the output of D_1. Discriminator D_2 is an adversarial regularizer and tells the difference between the generated group features $\hat{X}_{1:K}$ and group features of full videos $F_{1:K}(X)$. Using the adversarial loss, SRAM is encouraged to generate features that are indistinguishable from the group features of the corresponding full videos:

$$\begin{aligned}\mathcal{L}_{\text{GAN}} =& \mathbb{E}_{X_{(1:T)}\sim p_{\text{data}}(X_{(1:T)})} \log D_2\left(F_{1:K}(X)\right) \\ &+ \mathbb{E}_{X_{(1:t_0)}\sim p_{\text{data}}(X_{(1:t_0)})} \log\left(1 - D_2(\hat{X}_{1:K})\right).\end{aligned} \tag{3.7}$$

Note that the generated group representation $\hat{X}_{1:K}$ is computed by SRAM S from the partial observation $X_{1:t_0}$.

Sequential reconstruction loss is proposed to align the predicted activity representations to become close to the ground-truth activity representations at each unrolling stage. Since our method has K-stage sequential prediction, it is necessary to encourage the predicted group representations $\hat{X}_{1:K}$ on each of the K unrolling stages to become close to the ground-truth features at that timestamp. This is different from adversarial loss that only align the generated features to be close to ground-truth at full observation stage.

We train a separate ST-GCN $F(\cdot)$ as a recognition model to obtain the group activity representations of full videos X for training. The resulting frame-wise group representations $F(X)$ are used to encourage the activity features of the i-th person generated at the k-th unrolling stage to be similar to the ground-truth features using

$$\mathcal{L}_{\text{rec}} = \frac{1}{K \times N} \sum_{k=1}^{K} \sum_{i=1}^{N} ||\hat{\mathbf{x}}_k(i) - F_k(X, i)||^2 . \tag{3.8}$$

Here, $F_k(X, i)$ is the features of the i-th person of the full video at the k-th stage. This loss function sequentially computes the difference between the predicted features $\hat{\mathbf{x}}_k(i)$ (the i-th row on $\hat{X}_{1:K}$) for the i-th person at unrolling stage k and the ground-truth features $F_k(X, i)$, mimicking how a partial observation is progressively approaching its corresponding full observation.

Position regression loss. We use the tracklets of individuals provided by Ibrahim and Mori (2018) as the ground-truth of individuals positions. During training, we use the mean square error between the predicted positions and ground-truth positions at the K unrolling stages as loss function:

$$\mathcal{L}_{\text{reg}} = \frac{1}{K \times N} \sum_{k=1}^{K} \sum_{i=1}^{N} ||\hat{\mathbf{b}_k}(i) - \mathbf{b}_k(i)||^2, \tag{3.9}$$

where the predicted position $\hat{\mathbf{b}_k}(i)$ is the i-th row of $\hat{B}_k$, *i.e.,* , the i-th person's position predicted by the sequential decoder at the k-th stage.

Model learning. During training, the overall objective function is written as a sum of sequential reconstruction loss $\mathcal{L}_{\text{rec}}$, adversarial loss $\mathcal{L}_{\text{GAN}}$, classification loss $\mathcal{L}_{\text{cls}}$ implemented by softmax loss, and position regression loss $\mathcal{L}_{\text{reg}}$:

$$\min_{\mathcal{E},\mathcal{D}} \max_{D_1,D_2} \mathcal{L}_{\text{rec}} + \mathcal{L}_{\text{GAN}} + \mathcal{L}_{\text{cls}} + \mathcal{L}_{\text{reg}}. \tag{3.10}$$

Sequential relational anticipation model $(\mathcal{E}, \mathcal{D})$ and two discriminators (D_1, D_2) are alternatively trained until convergence.

3.3.6 Discussion

Group activity modeling and anticipation. Our SRAM captures the interactions of multiple people in the observation encoder, and anticipates their future relations by a sequential decoder. This is different from existing action prediction methods Kong et al. (2020); Qi et al. (2018) that can only predict the action of an individual. We believe such a novel method will pave the way for future research in other structured visual prediction.

Structured sequential prediction. Compared with group activity recognition methods Wu et al. (2019); Qi et al. (2018); Ibrahim and Mori (2018), our method performs *sequential prediction* of group activity, in form of future positions and activity representations. Our activity prediction is also facilitated by explicitly predicting people's future positions.

Activity evolution over time. Our sequential decoder *progressively* predicts future representations through several unrolling stages, which boosts performance when only few frames are observed. It is guided by a sequential reconstruction loss, mimicking how a partial observation is sequentially approaching its full observation and an adversarial loss to make the generated full observation features to become indistinguishable from the real full observation features.

3.4 Group Activity Prediction Evaluation

3.4.1 Datasets

Volleyball Dataset Ibrahim et al. (2016) consists of 4830 video clips distributed in 8 group activities, such as *left spiking* and *right setting*. Each clip has 41 frames. Ibrahim et al. (2016) provides the players' tracklets and splits the dataset into training, validation and testing sets. Existing group activity recognition methods Wu et al. (2019); Ibrahim et al. (2016); Ibrahim and Mori (2018); Qi et al. (2018) use the middle 10 frames of each video. To generalize it to prediction task, we extend it to use the middle 20 frames as full observations, in order to model sequential dynamics. Note that the middle 20 frames contain complete group activity executions, because athletes generally move quickly to complete a group activity, such as direct spiking in a volleyball game.

Collective Activity Dataset (CAD) Choi et al. (2009) contains 44 videos with 5 group activities, including *crossing*, *queueing*, *walking*, *talking* and *waiting*. The group activities in CAD are labeled as the majority of people's individual actions. We use the existing tracklet information and training/testing splits following Wu et al. (2019). The number of the frames in videos ranges from 100 to 2000. Following Qi et al. (2018); Wu et al. (2019); Bagautdinov et al. (2017), we divide each video into 10-frame video clips. This expands training and testing data to 1746 and 765 clips, respectively. CAD mainly contains periodic activities such as *walking*, in which significant changes can be seen in 10 frames.

3.4.2 Implementation Details

Following Wu et al. (2019), we extract a 1024-dimensional feature vector for each individual with tracklets provided by Ibrahim et al. (2016), using Inception-v3 Szegedy et al. (2016) as backbone and ROIAlign He et al. (2017). We use three steps for training: First, Inception-v3 pretrained on ImageNet is fine-tuned on single frames by jointly predicting individual actions and group activities. Then, we freeze the backbone and finetune the recognition model $F(\cdot)$ given full videos in the training set. The recognition model contain two ST-GCN layers Yan et al. (2018b), both with 256-d hidden units. After that, we train the proposed model. The observation encoder has two layers ST-GCN with both 256-d hidden units. The activity auto-encoder's encoder has one graph convolution layer that encodes the input into 256-d latent feature space. The position auto-encoder has two-layer graph convolution by encoding the 2-d positions into 64-d space and then 256-d latent space. During training, SRAM plus classifier D_1 and discriminator D_2 are alternatively updated.

The experiments are conducted with 10 different observation ratios ranging from 10% to 100% of full videos length. The number of unrolling stages K is set to 5. We use stochastic gradient descent for optimization. For Volleyball dataset, the three steps are trained for 30 epochs, 10 epochs and 20 epochs with learning rate 0.001, 0.001, 0.0001 respectively. For Collective Activity Dataset, the three steps are trained for 20 epochs, 50 epochs and 10 epochs with learning rate 0.0001, 0.0001, 0.0005, respectively.

3.4.3 Comparison with State-of-the-art

We compare our method with the state-of-the-art prediction methods LRCN Tran et al. (2015), DeepSCN Kong et al. (2017), IBoW and DBoW Ryoo (2011), KD Wang et al. (2019), AAPNet Kong et al. (2020) and state-of-the-art group activity recognition methods, including HRN Ibrahim and Mori (2018), HDTM Ibrahim et al. (2016) ARG Wu et al. (2019), SSU Bagautdinov et al. (2017). Following these methods' original setting, LRCN and HDTM adopt the AlexNet Krizhevsky et al. (2012) as the backbone. HRN, IBow, DBow, DeepSCN and original AAPNet use VGG-19. Our

Models	Tracklet	Backbones	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
LRCN	No	AlexNet	48.17	51.61	54.67	57.44	59.76	61.23	63.75	64.32	64.77	65.37
HDTM	Yes	AlexNet	52.43	59.09	66.04	76.37	80.48	81.82	84.07	84.47	84.60	84.06
IBoW	No	VGG	58.03	60.72	64.84	65.26	67.51	70.80	73.45	74.24	74.29	75.63
DBoW	No	VGG	58.03	55.56	56.16	58.93	59.90	61.97	63.79	63.06	63.88	64.78
DeepSCN	No	VGG	59.46	62.23	65.52	70.38	72.55	77.37	79.75	80.35	80.31	80.78
HRN	Yes	VGG	52.58	56.99	64.32	74.49	76.96	80.36	83.72	84.74	84.08	85.30
KD	No	VGG	65.67	67.68	70.00	70.83	71.96	72.10	73.22	73.30	73.30	73.90
AAPNet	No	VGG	59.53	65.37	68.29	72.25	75.24	77.79	79.91	80.25	80.18	80.78
e-AAPNet	Yes	InceptionV3	62.98	70.31	77.64	83.55	84.91	85.86	87.54	87.23	87.92	89.01
SSU	Yes	InceptionV3	63.20	70.65	79.66	84.07	87.13	87.65	88.30	88.18	88.41	89.01
ARG	Yes	InceptionV3	64.82	69.41	76.07	79.43	82.70	83.99	85.04	85.19	85.86	85.94
Ours	Yes	InceptionV3	**77.86**	**82.57**	**84.97**	**87.06**	**88.63**	**88.93**	**89.08**	**88.93**	**88.48**	**91.97**

Table 3.1 **Group activity prediction accuracy (%) on Volleyball dataset with observation ratios ranging from** 10% **to** 100%. Group activity recognition results can be seen from the last column, in which 100% frames are observed.

method follows ARG and SSU to use Inception-V3 method. HRN, HDTM, ARG, SSU and our method adopt the tracklets of players provided by Ibrahim et al. (2016). To make a fair comparison, we extend state-of-the-art action prediction method AAPNet (“e-AAPNet” for simplification) to make use of tracking information and use Inception-V3 as backbone. We train all of the comparison methods using the parameters described in their original papers.

3.4.3.1 Results on Volleyball dataset.

Table 3.1 summarizes the prediction performance of the proposed method, existing action prediction methods and group activity recognition methods. Results demonstrate that our model outperforms the comparison methods. Existing action prediction methods, e.g., LRCN, IBoW, DBoW, DeepSCN, AAPNet, KD propose to improve the prediction performance by information transfer. However, they regard multiple people as a single entity and do not consider the interactions between multiple people. Thus, the extracted features do not contain informative cues of the interactions of people, resulting in a low prediction performance. The proposed method uses tracklets Ibrahim et al. (2016), while the existing predictors for individuals e.g. LRCN, IBoW, DBoW, DeepSCN, AAPNet, KD do not. To make a fair comparison, we extend AAPNet to use tracklet information. Experimental results show that our method can predict the dynamics of interactions and better enrich partial observations.

Models	Tracklet	50%	100%
ARG Wu et al. (2019)	Yes	88.10	88.37
DeepSCN Kong et al. (2017)	No	81.31	82.22
AAPNet Kong et al. (2020)	No	81.57	82.75
e-AAPNet Kong et al. (2020)	Yes	86.01	86.67
Ours	Yes	**92.55**	**92.81**

Table 3.2 Prediction accuracy (%) on Collective Activity Dataset with observation ratios 50% and 100%.

Group activity recognition methods such as HDTM, SSU, HRN, ARG do not have capability of gaining extra information from full activity executions. Thus, when the observation ratio is very low (10% or 20% observations), their performance is much lower than our method. Note that ARG applies random sampling strategy by sampling three frames from an entire video as input. In the comparison experiment, this strategy is applied in each of the partial observations as input. The proposed method consistently outperforms ARG, as our method captures the temporal dynamics of multiple people in the group, and sequentially generates features close to the corresponding full observations. It improves the representation power of the partial observations, and facilitates group activity prediction.

3.4.3.2 Results on Collective Activity Dataset.

Comparison results are listed in Table 3.2. Our method outperforms existing methods ARG, DeepSCN, and AAPNet by a large margin. Given tracklets as input, our method is 6.54% higher than e-AAPNet at 50% observation ratio since the people's actions and relations are predicted in our model. Group activities such as *group walking* are cyclic, and thus the prediction performance of our method at 50% observation ratio is close to the one at 100% observation ratio.

3.4.4 Ablation Study

We perform detailed ablation studies on the Volleyball dataset to evaluate the contributions of the sequential prediction strategy, as well as the loss functions.

3.4.4.1 How much does the prediction loss help?

The impacts of loss functions are analyzed on Volleyball dataset in detail. The evaluation results can also validate the contributions of the proposed sequential prediction strategy. We compare the following the variants, including: (**1**) without the position regression loss $\mathcal{L}_{\text{reg}}$ defined in Eq. (3.9).

In this variant, the position auto-encoder for predicting future positions is not used. During sequential prediction, we replace the individuals' positions in the future frames by the ones given by the last observed frame's. The positions are used for computing G^{p} for each unrolling stage. (**2**) without adversarial loss $\mathcal{L}_{\mathrm{GAN}}$. (**3**) without sequential reconstruction loss $\mathcal{L}_{\mathrm{rec}}$ for generated features of unrolling stages. (**4**) The proposed full network.

Compared with variant (**1**), the significant performance gains with all different observation ratios show that the prediction of people's positions is of high importance for group activity prediction. Compared with variants (**2**) and (**3**), it shows that the adversarial loss $\mathcal{L}_{\mathrm{GAN}}$ and the reconstruction loss $\mathcal{L}_{\mathrm{rec}}$ in our method improve the performance by 0.55% and 0.63% on average, respectively. Therefore, the proposed sequential decoder guided by $\mathcal{L}_{\mathrm{GAN}}$ and $\mathcal{L}_{\mathrm{rec}}$ can generate more discriminative activity representations at each stage.

3.4.4.2 How much does the sequential prediction help?

Our sequential decoder predicts group activity representations at K unrolling stages. In this experiment, we evaluate the effect of the number of unrolling stages K on the prediction performance. We set K to 1, 2, 5, and 10, and compare the prediction performance. Table 3.3 indicates that the best overall prediction performance is achieved when $K = 5$. The prediction performance is slightly affected when $K = 10$, but the computational complexity of the prediction model is increased due to the extra unrolling stages. The average prediction performance drops to 81.47% if $K = 1$. The variant with $K = 1$ is the one that directly maps partial observation in one unrolling stage, similar to what Kong et al. (2020, 2017) do. The result demonstrates the superiority of our progressive prediction in anticipating discriminative group representations given partial observations. If more stages are allowed ($K = 5$ or $K = 10$), the sequential decoder in our model can progressively generate discriminative features for group activity prediction even though it is given very limited frames. Therefore, its prediction performance is improved.

Loss	10%	40%	70%	Average
(1)$\mathcal{L}_{\text{GAN}}$+$\mathcal{L}_{\text{rec}}$+$\mathcal{L}_{\text{cls}}$	75.09	85.59	87.06	84.85
(2) $\mathcal{L}_{\text{rec}}$+$\mathcal{L}_{\text{reg}}$+$\mathcal{L}_{\text{cls}}$	76.14	85.79	88.18	86.30
(3) $\mathcal{L}_{\text{reg}}$+$\mathcal{L}_{\text{GAN}}$+$\mathcal{L}_{\text{cls}}$	77.61	83.22	85.64	86.22
(4) Ours	**77.86**	**87.67**	**89.08**	**86.85**

(a)

K	10%	40%	70%	Average
1	70.38	80.02	86.14	81.47
2	72.36	86.59	89.07	85.22
5	77.86	**87.06**	89.08	**86.85**
10	**77.93**	86.69	**89.23**	86.79

(b)

Table 3.3 Ablation studies on volleyball dataset. We show the accuracy(%) given videos of observation ratio at 10%, 40%, 70%.

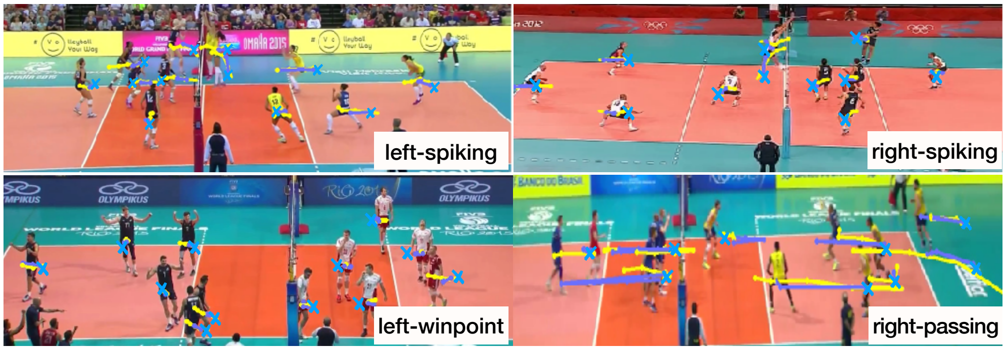


Figure 3.4 Visualization of position predictions. The blue and the yellow lines denote the prediction positions and ground-truth positions Ibrahim et al. (2016), respectively. “×” indicates the starting point of the movement. Best viewed in color.

3.4.5 Position Prediction Evaluation

3.4.5.1 Visualization of predicted positions

As shown in Fig. 3.4, we visualize the movement of individuals learned by the position auto-encoder in SRAM. The position auto-encoder progressively predicts the positions of individuals at the unrolling stages. The visualization result shows our position auto-encoder can successfully predict the directions and step sizes of individuals in the future based on partial observations. Although Fig. 3.4 (bottom-right) shows the direction of the predicted movement is mostly accurate, the future position of a person is not accurate if the person moves very fast.

3.4.5.2 Quantitative evaluation

We quantitatively evaluate our position prediction results compared to two popular trajectory prediction methods SocialGAN Gupta et al. (2018) and SocialLSTM Alahi et al. (2016). Following SocialGAN, Final Displacement Error (FDE) is used to compute the Euclidean distance between the predicted positions and ground-truth positions at the final timestamp and Average Displacement Error (ADE) is used to compute that at each unrolling stage.

Method	FDE	ADE
SocialGAN Gupta et al. (2018)	5.32	3.05
SocialLSTM Alahi et al. (2016)	6.44	4.44
Ours	**3.62**	**2.44**

Table 3.4 Final Displacement Error (FDE) and Average Displacement Error (ADE) for position prediction.

As shown in Tab. 3.4, the results demonstrate that our method can accurately predict the future positions for a group of people, and our method outperforms the two trajectory prediction methods. This is mainly because we capture the relational action dynamics of multiple people while SocialGAN and SocialLSTM do not.

3.5 Summary

We have proposed a novel sequential relational anticipation model (SRAM) to predict group activity given only the beginning frames of an activity execution. Our model captures complex relational dynamics of multiple people in the observed frames. It then anticipates the group representations including group activity features and position features. A novel sequential decoder is proposed to progressively anticipate the group representations through several unrolling stages. Extensive results on two datasets demonstrate that our method significantly outperforms the state-of-the-art methods. Results also validate that the progressive anticipation using multiple unrolling stages facilitates group activity prediction. Further experimental results show that the modeling and prediction of people's positions improve our performance on group activity prediction.

BIBLIOGRAPHY

Alahi, A., Goel, K., Ramanathan, V., Robicquet, A., Fei-Fei, L., and Savarese, S. (2016). Social lstm: Human trajectory prediction in crowded spaces. In *CVPR*, pages 961–971.

Amer, M. R., Lei, P., and Todorovic, S. (2014). Hirf: Hierarchical random field for collective activity recognition in videos. In *ECCV*, pages 572–585.

Azar, S. M., Atigh, M. G., Nickabadi, A., and Alahi, A. (2019). Convolutional relational machine for group activity recognition. In *CVPR*, pages 7892–7901.

Bagautdinov, T., Alahi, A., Fleuret, F., Fua, P., and Savarese, S. (2017). Social scene understanding: End-to-end multi-person action localization and collective activity recognition. In *CVPR*, pages 4315–4324.

Biswas, S. and Gall, J. (2018). Structural recurrent neural network (srnn) for group activity analysis. In *WACV*, pages 1625–1632.

Cai, Y., Li, H., Hu, J.-F., and Zheng, W.-S. (2019). Action knowledge transfer for action prediction with partial videos. pages 8118–8125.

Choi, W., Shahid, K., and Savarese, S. (2009). What are they doing?: Collective activity classification using spatio-temporal relationship among people. In *ICCV Workshops*, pages 1282–1289.

Deng, Z., Vahdat, A., Hu, H., and Mori, G. (2016). Structure inference machines: Recurrent neural networks for analyzing relations in group activity recognition. In *CVPR*, pages 4772–4781.

Gammulle, H., Denman, S., Sridharan, S., and Fookes, C. (2018). Multi-level sequence gan for group activity recognition. In *Asian Conference on Computer Vision*, pages 331–346.

Gammulle, H., Denman, S., Sridharan, S., and Fookes, C. (2019). Predicting the future: A jointly learnt model for action anticipation. In *ICCV*, pages 5562–5571.

Gavrilyuk, K., Sanford, R., Javan, M., and Snoek, C. G. M. (2020). Actor-transformers for group activity recognition. In *CVPR*.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative adversarial nets. In *NIPS*, pages 2672–2680.

Gupta, A., Johnson, J., Fei-Fei, L., Savarese, S., and Alahi, A. (2018). Social gan: Socially acceptable trajectories with generative adversarial networks. In *CVPR*, pages 2255–2264.

He, K., Gkioxari, G., Dollár, P., and Girshick, R. (2017). Mask R-CNN. In *ICCV*.

Hu, G., Cui, B., He, Y., and Yu, S. (2020). Progressive relation learning for group activity recognition.

Hu, J.-F., Zheng, W.-S., Ma, L., Wang, G., Lai, J.-H., and Zhang, J. (2018). Early action prediction by soft regression. *TPAMI*.

Ibrahim, M. S. and Mori, G. (2018). Hierarchical relational networks for group activity recognition and retrieval. In *ECCV*, pages 721–736.

Ibrahim, M. S., Muralidharan, S., Deng, Z., Vahdat, A., and Mori, G. (2016). A hierarchical deep temporal model for group activity recognition. In *CVPR*, pages 1971–1980.

Kipf, T. N. and Welling, M. (2017). Semi-supervised classification with graph convolutional networks. In *International Conference on Representation Learning (ICLR)*.

Kong, Y., Gao, S., Sun, B., and Fu, Y. (2018). Action prediction from videos via memorizing hard-to-predict samples. In *AAAI*.

Kong, Y., Kit, D., and Fu, Y. (2014). A discriminative model with multiple temporal scales for action prediction. In *ECCV*, pages 596–611.

Kong, Y., Tao, Z., and Fu, Y. (2017). Deep sequential context networks for action prediction. In *CVPR*, pages 1473–1481.

Kong, Y., Tao, Z., and Fu, Y. (2020). Adversarial action prediction networks. *TPAMI*.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *NIPS*, pages 1097–1105.

Lan, T., Chen, T.-C., and Savarese, S. (2014). A hierarchical representation for future action prediction. In *ECCV*, pages 689–704.

Lan, T., Sigal, L., and Mori, G. (2012). Social roles in hierarchical models for human activity recognition. In *CVPR*, pages 1354–1361.

Lea, C., Flynn, M. D., Vidal, R., Reiter, A., and Hager, G. D. (2017). Temporal convolutional networks for action segmentation and detection. In *CVPR*, pages 156–165.

Li, X. and Choo Chuah, M. (2017). Sbgar: Semantics based group activity recognition. In *ICCV*, pages 2876–2885.

Ma, S., Sigal, L., and Sclaroff, S. (2016). Learning activity progression in lstms for activity detection and early detection. In *CVPR*, pages 1942–1950.

Martinez, J., Black, M. J., and Romero, J. (2017). On human motion prediction using recurrent neural networks. In *CVPR*, pages 2891–2900.

Qi, M., Qin, J., Li, A., Wang, Y., Luo, J., and Van Gool, L. (2018). stagnet: An attentive semantic rnn for group activity recognition. In *ECCV*, pages 101–117.

Ramanathan, V., Huang, J., Abu-El-Haija, S., Gorban, A., Murphy, K., and Fei-Fei, L. (2016). Detecting events and key actors in multi-person videos. In *CVPR*, pages 3043–3053.

Ryoo, M. S. (2011). Human activity prediction: Early recognition of ongoing activities from streaming videos. In *ICCV*, pages 1036–1043.

Sadegh Aliakbarian, M., Sadat Saleh, F., Salzmann, M., Fernando, B., Petersson, L., and Andersson, L. (2017). Encouraging lstms to anticipate actions very early. In *ICCV*, pages 280–289.

Shi, Y., Fernando, B., and Hartley, R. (2018). Action anticipation with rbf kernelized feature mapping rnn. In *ECCV*, pages 301–317.

Shu, T., Todorovic, S., and Zhu, S.-C. (2017). Cern: confidence-energy recurrent network for group activity recognition. In *CVPR*, pages 5523–5531.

Soomro, K., Zamir, A. R., and Shah, M. (2012). Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*.

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., and Wojna, Z. (2016). Rethinking the inception architecture for computer vision. In *CVPR*, pages 2818–2826.

Tang, Y., Wang, Z., Li, P., Lu, J., Yang, M., and Zhou, J. (2018). Mining semantics-preserving attention for group activity recognition. In *Multimedia*, pages 1283–1291.

Tran, D., Bourdev, L., Fergus, R., Torresani, L., and Paluri, M. (2015). Learning spatiotemporal features with 3d convolutional networks. In *ICCV*.

Vondrick, C., Pirsiavash, H., and Torralba, A. (2016). Anticipating visual representations from unlabeled video. In *CVPR*, pages 98–106.

Wang, M., Ni, B., and Yang, X. (2017). Recurrent modeling of interaction context for collective activity recognition. In *CVPR*, pages 3048–3056.

Wang, X., Hu, J.-F., Lai, J.-H., Zhang, J., and Zheng, W.-S. (2019). Progressive teacher-student learning for early action prediction. In *CVPR*, pages 3556–3565.

Wang, Z., Shi, Q., Shen, C., and Van Den Hengel, A. (2013). Bilinear programming for human activity recognition with unknown mrf graphs. In *CVPR*, pages 1690–1697.

Wichers, N., Villegas, R., Erhan, D., and Lee, H. (2018). Hierarchical long-term video prediction without supervision. pages 6038–6046.

Wu, J., Wang, L., Wang, L., Guo, J., and Wu, G. (2019). Learning actor relation graphs for group activity recognition. In *CVPR*, pages 9964–9974.

Yan, R., Tang, J., Shu, X., Li, Z., and Tian, Q. (2018a). Participation-contributed temporal dynamic model for group activity recognition. In *Multimedia*, pages 1292–1300.

Yan, S., Xiong, Y., and Lin, D. (2018b). Spatial temporal graph convolutional networks for skeleton-based action recognition. In *AAAI*.

Yan, Y., Ni, B., and Yang, X. (2017). Predicting human interaction via relative attention model. *IJCAI*, pages 3245–3251.

Yao, T., Wang, M., Ni, B., Wei, H., and Yang, X. (2018). Multiple granularity group interaction prediction. In *CVPR*, pages 2246–2254.

Zhao, H. and Wildes, R. P. (2019). Spatiotemporal feature residual propagation for action prediction. In *ICCV*, pages 7003–7012.

CHAPTER 4

GATED HISTORY UNIT WITH BACKGROUND SUPPRESSION FOR ONLINE ACTION DETECTION

4.1 Introduction

Online action detection is the task to predict actions in a streaming video as they unfold De Geest et al. (2016). It is critical to applications including autonomous driving, public safety, virtual and augmented reality. Unlike action detection in the offline setting, where the entire untrimmed video is observable at any given moment, a major challenge for online action detection is that the predictions are solely based on observations of history without access to video frames in the future. The model needs to build a causal reasoning of the present in correlation to what happened hitherto, and as efficiently as possible for the online setting.

Prior work for online action detection Xu et al. (2019); Eun et al. (2020, 2021); Gao et al. (2021); Qu et al. (2020); Zhao et al. (2020) includes recurrent-based LSTMs Hochreiter and Schmidhuber (1997) and GRUs Cho et al. (2014) that are prone to forgetting informative history as sequential frame processing is ineffective in preserving long-range interactions. Emerging methods Wang et al. (2021b); Xu et al. (2021a) employ transformers Vaswani et al. (2017) to mitigate this by encoding sequential frames in parallel via self-attention. Some improve model efficiency by using cross-attention Xu et al. (2021a); Jaegle et al. (2021a) to compress the video sequence into a fixed-sized latent encoding for prediction.

Fig. 4.1 shows an example video stream (middle row) where the latest (current) frame contains *Cliff Diving* action. It is worth noting that, as commonly observed in video sequences, not every history frame is informative for current frame prediction (*e.g.,* frames showing people cheering or camera panning in Fig. 4.1). Existing transformer-based approaches Xu et al. (2021a) use vanilla cross-attention to learn attention weights for history frames that determine their contribution to the current frame prediction. Such attention weights do not correlate with how informative each history frame is to current frame prediction. As shown in Fig. 4.1 (top row), when history frames are ordered from lower to higher cross-attention weights for vanilla cross-attention, frames that are

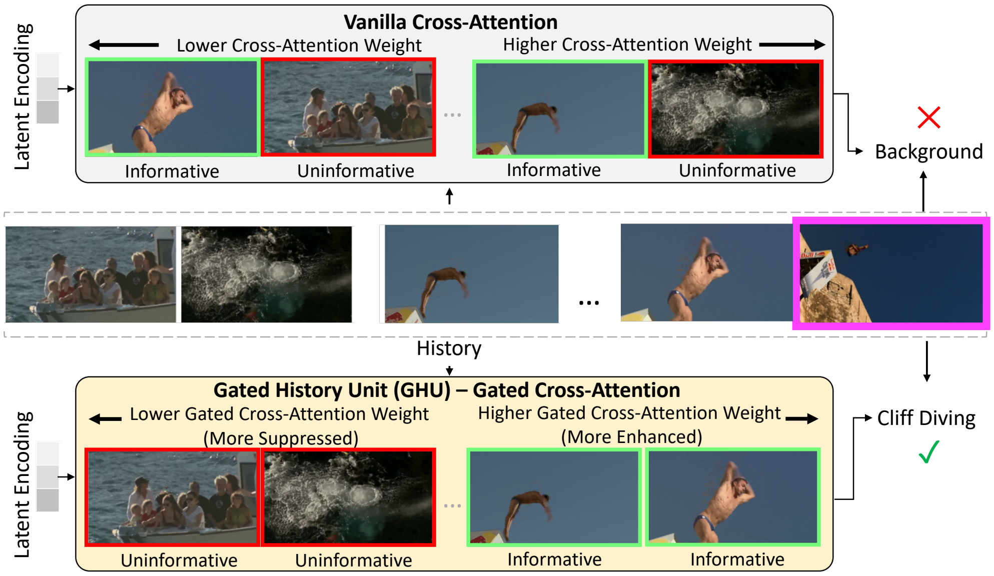


Figure 4.1 We show an example video stream (middle row) where the current frame (magenta) contains *Cliff Diving* action. Weights from vanilla cross-attention (top row) do not correlate with how informative each history frame is to current frame prediction, leading to incorrect prediction of *Background*. Our novel Gated History Unit (GHU) (bottom row) calibrates cross-attention weights using gating scores to enhance history frames that are informative to current frame prediction (green) and suppress uninformative ones (red), leading to accurate prediction of *Cliff Diving*.

informative for current frame prediction may have lower weights while uninformative frames may have higher weights, leading to incorrect current frame prediction. Another common challenge for existing methods is false positive prediction for background frames that closely resemble action frames (*e.g.*, pre-shot routine before golf swing). Existing methods also do not leverage that although future frames are not available for the current frame prediction, subsequently observed frames that are future to the history can be leveraged to enhance history encoding, which in return improves current frame prediction.

To address the above limitations, we propose GateHUB, **Gate**d **H**istory **U**nit with **B**ackground suppression. GateHUB comprises a novel Gated History Unit (GHU), a position-guided gated cross-attention module that enhances informative history while suppressing uninformative frames via gated cross-attention (as shown in Fig. 4.1, bottom row). GHU enables GateHUB to encode more informative history into the latent encoding to better predict for current frame. GHU combines the benefit of an LSTM-inspired gating mechanism to filter uninformative history with the transformer's ability to effectively learn from long sequences.

GateHUB leverages *future frames for history* by introducing Future-augmented History (FaH). FaH extracts features for a history frame using its future, *i.e.,* the subsequently *observed* frames. This makes a history frame aware of its future and helps it to be more informative for current frame prediction. To tackle the common false positives in prior art, GateHUB proposes a novel background suppression objective that has different treatments for low-confident action and background predictions. These novel approaches enable GateHUB to outperform all existing methods on common benchmark datasets THUMOS Idrees et al. (2017), TVseries De Geest et al. (2016), and HDD Ramanishka et al. (2018a). Keeping model efficiency in mind for the online setting, we also validate that GateHUB is more efficient than the existing best method Xu et al. (2021a) while being more accurate. Moreover, our proposed optical flow-free variant is 2.8× faster than all existing methods that require both RGB and optical flow data with higher or close accuracy.

To summarize, our main contributions are:

1. Gated History Unit (GHU), a novel position-guided gated cross-attention that explicitly enhances or suppresses parts of video history as per how informative they are to predicting action for the current frame.

2. Future-augmented History (FaH) to extract features for a history frame using its subsequently observed frames to enhance history encoding.

3. A background suppression objective to mitigate the false positive prediction of background frames that closely resemble the action frames.

4. GateHUB is more accurate than all existing methods and is also more efficient than the existing best work. Moreover, our proposed optical flow-free model is 2.8× faster compared to all existing methods that require both RGB and optical flow information while achieving higher or close accuracy.

4.2 Related Work

Online Action Detection. Previous methods for online action detection include use 3D ConvNet De Geest et al. (2016), reinforcement learning Gao et al. (2017a), recurrent networks Xu et al. (2019); Eun et al. (2020); Qu et al. (2020); Gao et al. (2021); Zhao et al. (2020); Qu et al. (2020) and more recently, transformers Wang et al. (2021b); Xu et al. (2021a). The primary challenge in leveraging history is that for long untrimmed videos, its length becomes intractably long over time. To make it computationally feasible, some Eun et al. (2020); Wang et al. (2021b); Gao et al. (2021); Qu et al. (2020) make the online prediction conditioned only on the most recent frames spanning less than a minute. This way the history beyond this duration that might be informative to current frame predictions is left unused. TRN Xu et al. (2019) mitigates this by the hidden state in LSTMs Hochreiter and Schmidhuber (1997) to memorize the entire history during inference. But LSTM limits its ability to model long-range temporal interactions. More recently, Xu et al. (2021a) proposes to scale transformers to the history spanning longer duration. However, not every history frame is informative and useful. Xu et al. (2021a) lacks the forgetting mechanism of LSTM to filter uninformative history which causes it to encode uninformative history into the encoding leading to incorrect predictions. Our Gated History Unit (GHU) and Future-augmented History (FaH) combine the benefits of LSTM's selective encoding and transformer's long-range modeling to leverage long-duration history more informatively to outperform all previous methods.

Transformers for Video Understanding. Transformers can achieve superior performance on video understanding tasks by effectively modeling the spatiotemporal context via attention. Most of the previous transformer-based methods Bertasius et al. (2021); Arnab et al. (2021); Fan et al. (2021); Neimark et al. (2021) focus on action recognition in trimmed videos Carreira and Zisserman (2017) (videos spanning few seconds) due to the quadratic complexity *w.r.t.* video length. Untrimmed videos have a longer duration from a few minutes to hours and contain frames with irrelevant actions (labeled as background). Temporal action localization (TAL) Shou et al. (2016); Xu et al. (2017); Gao et al. (2017b); Shou et al. (2017); Zhao et al. (2017); Buch et al. (2017); Liu et al. (2019); Lin et al. (2019); Zhu et al. (2021); Zhang et al. (2021) and temporal action proposal generation (TAP) Lin et al. (2018, 2019); Tan et al. (2021) are two fundamental tasks in untrimmed video understanding. AGTNawhal and Mori (2021) proposes activity graph transformer for TAL based on DETR Carion et al. (2020). TAPGWang et al. (2021a) applies transformer to predict the activity boundary for TAP. However, unlike TAL and TAP which are both offline tasks having access to the entire video, online action detection does not have access to the future and requires causal understanding from history to present. We follow the existing transformer-based streaming tasksGirdhar and Grauman (2021); Chen et al. (2021); Xu et al. (2021a) and apply a causal mask to address online action detection.

Long Sequence Modeling. To model long input sequences, recent work Dosovitskiy et al. (2021) proposes to reduce complexity by factorizing Touvron et al. (2021) or subsampling the inputs Chen et al. (2020). Another group of work focuses on modifying the internal dense self-attention module to boost the efficiency Beltagy et al. (2020); Wang et al. (2020). More recently, Perceiver Jaegle et al. (2021b) and PerceiverIO Jaegle et al. (2021a) propose to cross-attend long-range inputs to a small fixed-sized latent encoding, adding further flexibility in terms of input and reducing the computational complexity. However, unlike our GHU, PerceiverIO lacks an explicit mechanism to enhance/suppress history frames making it sub-optimal for online action detection. Our method uses LSTM-inspired gating to calibrate cross-attention to enhance/suppress history frames per their informative-ness while employing transformers to learn from long history sequences effectively.

4.3 Methodology

Given a streaming video sequence $\mathbf{h} = [h_t]_{t=-T+1}^{0}$, our task is to identify *if* and *what* action $y_0 \in \{0, 1, ..., C\}$ occurs at the current frame h_0. We have a total of C action classes and label 0 for background frames with no action. Since future frames $h_1, h_2, ...,$ are NOT accessible, the model makes the $C + 1$-way prediction for the current frame based on the recent T frames, $[h_t]_{t=-T+1}^{0}$, observed up until the current frame. While T may be large in an untrimmed video stream, as shown in the top row of Fig. 4.1, all frames observed in past history $[h_t]_{t=-T+1}^{-1}$ may not be equally informative to the prediction for the current frame.

4.3.1 Gated History Unit based History Encoder

To make the $C + 1$-way prediction accurately for current frame h_0 based on T history frames, $\mathbf{h} = [h_t]_{t=-T+1}^{0}$, we employ transformers to first encode the video sequence history and then

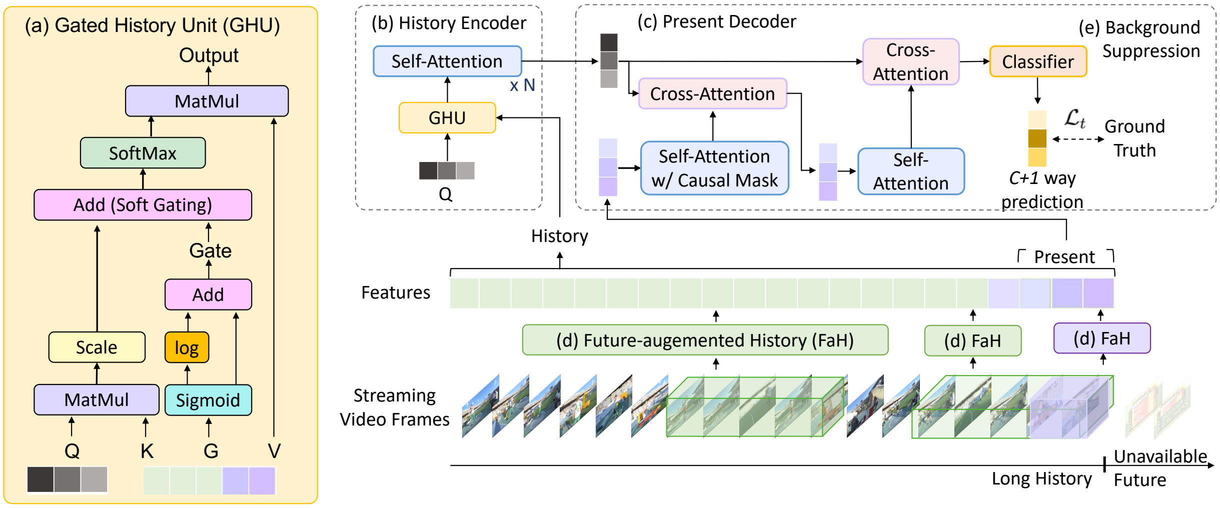


Figure 4.2 **Model Overview.** GateHUB comprises a novel Gated History Unit (GHU) (a) as part of History Encoder (b) to explicitly enhance or suppress history frames, *i.e.,* streaming video frames observed so far, as per how informative they are to current frame prediction. GHU encodes them by cross-attending with a latent encoding (Q). GateHUB uses Future-augmented History features (FaH) (d) to encode each history frame using t_f subsequently observed future frames. The Present Decoder (c) correlates with history by cross-attending the encoded history with the present, *i.e.,* , a small set of most recent frames, to make current frame prediction. We subject the prediction to a background suppression loss (e) to reduce false positives by effectively separating action frames from closely resembling background frames.

associate the current frame with the encoding for prediction. Inspired by the recently introduced PerceiverIO Jaegle et al. (2021a), our method consists of a History Encoder (Fig. 4.2b) that uses cross-attention to project the variable length history to a fixed-length learned latent encoding. Using cross-attention is more efficient than using self-attention because its computational complexity is quadratic *w.r.t.* latent encoding size instead of the video sequence length which is typically orders of magnitude larger. This is crucial to developing a model for the online setting. However, as shown in Fig. 4.1, vanilla cross-attention, as used in PerceiverIO and LSTR Xu et al. (2021a), fails to learn attention weights for history frames that correlate with how informative each history frame is for h_0 prediction. We therefore introduce a novel Gated History Unit (GHU) (Fig. 4.2a) that has a position-guided gated cross-attention mechanism which learns a set of gating scores G to calibrate the attention weights to effectively enhance or suppress history frames based on how informative they are to current frame prediction.

Specifically, given $\mathbf{h} = [h_t]_{t=-T+1}^{0}$ as the streaming sequence of T history frames ending at current frame h_0, we encode $\mathbf{h}$ with a feature extraction backbone, u, followed by a linear encoding layer $\mathbf{E}$. We then subject the output to a learnable position encoding, $\mathbf{E_{pos}}$, relative to the current frame, h_0, to give $\mathbf{z^h} = u(\mathbf{h})\mathbf{E} + \mathbf{E_{pos}}$ where $u(\mathbf{h}) \in \mathbb{R}^{T \times M}$, $\mathbf{E} \in \mathbb{R}^{M \times D}$, $\mathbf{z^h} \in \mathbb{R}^{T \times D}$ and $\mathbf{E_{pos}} \in \mathbb{R}^{T \times D}$. M and D denote the dimensions of extracted features and post-linear encoding features, respectively. We also define a learnable latent query encoding, $\mathbf{q} \in \mathbb{R}^{L \times D}$, that we cross-attend with $\mathbf{h}$. Following the standard multi-headed cross-attention setup Jaegle et al. (2021b,a), let N_{heads} be the number of heads in GHU such that $Q_i = \mathbf{q}\mathbf{W}_i^q$, $K_i = \mathbf{z^h}\mathbf{W}_i^k$, $V_i = \mathbf{z^h}\mathbf{W}_i^v$ be the queries, keys and values, respectively, for each head $i \in \{1, \ldots, N_{heads}\}$ (Fig. 4.2a) where projection matrices $\mathbf{W}_i^q, \mathbf{W}_i^k \in \mathbb{R}^{D \times d_k}$ and $\mathbf{W}_i^v \in \mathbb{R}^{D \times d_v}$. We assign $d_k = d_v = D/N_{heads}$ in our set up Vaswani et al. (2017). Next, we obtain the position-guided gating scores, G, for $\mathbf{h}$ as,

$$\mathbf{z^g} = \sigma(\mathbf{z^h}\mathbf{W}^g) \tag{4.1}$$

$$G = \log(\mathbf{z^g}) + \mathbf{z^g} \tag{4.2}$$

where $\mathbf{W}^g \in \mathbb{R}^{D \times 1}$ is the matrix projecting each history frame to a scalar. $\mathbf{z^g} \in \mathbb{R}^{T \times 1}$ is a sequence of scalars for the history frames $\mathbf{h}$ after applying sigmoid σ. $G \in \mathbb{R}^{T \times 1}$ is the gating score sequence for history frames in GHU. By using $\mathbf{z^h}$ which already contains the position encoding, the gates are guided by the relative position of the history frame to the current frame h_0. As also shown in Fig. 4.2a, we now compute the gated cross-attention for each head, GHU_i, as,

$$GHU_i = \text{Softmax}\left(\frac{Q_i K_i^T}{\sqrt{d_k}} + G\right) V_i \tag{4.3}$$

and multi-headed gated cross-attention defined as,

$$\text{MultiHeadGHU}(Q, K, V, G) = \text{Concat}([GHU_i]_{i=0}^{N_{heads}})\mathbf{W^o} \tag{4.4}$$

where $\mathbf{W}^o \in \mathbb{R}^{D \times D}$ re-projects the attention output to D dimension. It is possible to define G separately for each head but in our method, we find sharing G across all heads to perform better (Sec. 4.4.4). From Eqn. 4.1 and 4.2, we can observe that each scalar in $\mathbf{z^g}$ lies in $[0, 1]$ due to sigmoid which implies that each gating score in G lies in $[-\inf, 1]$. This enables the softmax function in Eqn. 4.3 to calibrate the attention weight for each history frame by a factor in $[0, e]$

such that a factor in $[0, 1)$ suppresses a given history frame and a factor in $(1, e]$ enhances a given history frame. This provides an explicit ability to GHU to learn to calibrate the attention weight of a history frame based on how informative the history frame is for prediction of h_0. Unlike previous methods with relative position bias Liu et al. (2021); Dai et al. (2019), G is input-dependent and learns based on the history frame and its position *w.r.t.* current frame. This enables GHU to assess how informative each history frame is based on its feature representation and relative position from the current frame h_0. We feed the output of GHU to a series of N self-attention layers to obtain the final history encoding (Fig. 4.2b).

4.3.2 Hindsight is 2020: Future-augmented History

Existing methods Wang et al. (2021b); Xu et al. (2019, 2021a); Gao et al. (2021); Eun et al. (2020) extract features for each frame by feed-forwarding the frame and optionally, a small set of past consecutive frames through pretrained networks like TSN Wang et al. (2016) and I3D Carreira and Zisserman (2017). It is worth noting that although for current frame prediction its future is not available, for the history frames their *future* is accessible and this *hindsight* can potentially improve the encoding of history for current frame prediction. Existing methods do not have a mechanism to leverage this. This inspires us to propose a novel feature extraction scheme, Future-augmented History (FaH), where we aggregate observed future information into the features of a history frame to make it aware of its so far observable future. Fig. 4.2d illustrates the FaH feature extraction process. For a history frame h_t and a feature extraction backbone u, when t_f *future history frames* for h_t can be observed, FaH extracts features for h_t using a set of frames $[h_i]_{i=t}^{t+t_f}$ (*i.e.,* history frame itself and its subsequently observed t_f future frames). Otherwise, FaH extracts features for h_t using a set of frames $[h_i]_{i=t-t_{ps}}^{t}$ (*i.e.,* history frame itself and its past t_{ps} frames),

$$u(h_t) = \begin{cases} u([h_i]_{i=t-t_{ps}}^{t}) & \text{if } t > -t_f \\ u([h_i]_{i=t}^{t+t_f}) & \text{if } t <= -t_f \end{cases} \tag{4.5}$$

At each new time step with one more new frame getting observed, FaH will feed-forward through u twice to extract features for (1) the new frame using $[h_i]_{i=-t_{ps}}^{0}$ frames and (2) h_{-t_f} that is now eligible to aggregate future information using $[h_i]_{i=-t_f}^{0}$ frames (as shown in Fig. 4.2d purple and green cuboid respectively). Thus, FaH has the same time complexity as existing feature extraction methods. FaH does not trivially incorporate all available subsequently observed frames. Instead, it encodes only from a set of future frames that are the most relevant to a history frame (as we empirically explain later in Chapter 4.4.4).

4.3.3 Present Decoder

In order to correlate the present with history to make current frame prediction, we sample a subset of t_{pr} most recent history frames $[h_t]_{t=-t_{pr}-1}^{0}$ to model the present (*i.e.,* the most immediate context) for h_0 using the Present Decoder (Fig. 4.2c). After extracting the features via FaH, we apply a learnable position encoding, $\mathbf{E_{pos}^{pr}}$, to each of the t_{pr} frame features and subject them to a multi-headed self-attention with a causal mask. The causal mask limits the influence of only the preceding frames on a given frame. We then cross-attend the output from self-attention with the history encoding from the History Encoder. Inspired by Perceiver Jaegle et al. (2021b), we repeat this process twice and the self-attention does not need a causal mask the second time. Finally, we feed the output corresponding to each of t_{pr} frames to the classifier layer for prediction.

4.3.4 Background Suppression Objective

Existing online action detection methods Wang et al. (2021b); Xu et al. (2019, 2021a); Gao et al. (2021); Eun et al. (2020) apply standard cross entropy loss for $C+1$-way multi-label per-frame prediction. Standard cross entropy loss does not consider that the "no action" background class does not belong to any specific action distribution and is semantically different from the C action classes. This is because background frames can be anything from completely blank at the beginning of a video to closely resemble action frames without actually being action frames (*e.g.,* , aiming before making a billiards shot). The latter is a common cause for false positives in online action detection. In addition to the complex distribution of background frames, untrimmed videos suffer from a sharp data imbalance where background frames significantly outnumber action frames.

To tackle these challenges, we design a novel background suppression objective that applies separate emphasis on low-confident action and background predictions during training to increase the margin between action and background frames (Fig. 4.2e). Inspired by focal loss Lin et al.

(2017), our objective function, $\mathcal{L}_t$ for frame h_t is defined as,

$$\mathcal{L}_t = \begin{cases} -y_t^0(1 - p_t^0)^{\gamma_b} \log(p_t^0) & \text{if } y_t^0 = 1 \\ -\Sigma_{i=1}^C y_t^i (1 - p_t^i)^{\gamma_a} \log(p_t^i) & \text{otherwise} \end{cases} \tag{4.6}$$

where $\gamma_a, \gamma_b > 0$ enables low-confident samples to contribute more to the overall loss forcing the model to put more emphasis on correctly predicting these samples. Unlike original focal loss Lin et al. (2017), our background suppression objective specializes for online action detection by applying separate γ to action classes and background. This separation is necessary to distinguish the action classes that have a more constrained distribution from the background class whose distribution is more complex and unconstrained. Our objective is the first attempt in online action detection to put separate emphasis on low-confident hard action and background predictions.

4.3.5 Flow-free Online Action Detection

Existing methods Xu et al. (2019); Wang et al. (2021b); Eun et al. (2020) for online action detection use optical flow in addition to RGB to capture fine-grained motion among frames. Computing optical flow takes much more time than feature extraction or model inference and can be unrealistic for time-critical applications such as autonomous driving. This motivates us to develop an optical flow-free version of GateHUB that is able to achieve higher or close accuracy compared to existing methods without time-consuming optical flow estimation. To capture motion without optical flow using only RGB frames, we leverage multiple temporal resolutions using a spatiotemporal backbone such as TimeSformer Bertasius et al. (2021). We extract two feature vectors for a frame h_t by encoding a frame sequence sampled at a higher frame rate spanning a smaller time duration and another frame sequence sampled at a lower frame rate spanning a larger time duration. Similar to the setup using RGB and optical flow features, we concatenate the two feature vectors before feeding them to GateHUB.

4.4 Experiments

4.4.1 Datasets

Following existing online action detection work Wang et al. (2021b); Xu et al. (2019); Eun et al. (2020); Gao et al. (2017a); Xu et al. (2021a), we evaluate GateHUB on three common benchmark datasets – THUMOS'14, TVSeries, and HDD.

THUMOS'14 Idrees et al. (2017) consists of over 20 hours of sports video and is annotated with 20 actions. We follow prior work Wang et al. (2021b); Xu et al. (2019) and train on the validation set (200 untrimmed videos) and evaluate on the test set (213 untrimmed videos).

TVSeries De Geest et al. (2016) includes 27 episodes of 6 popular TV shows with a total duration of 16 hours. It is annotated with 30 real-world everyday actions, *e.g.,* open door, run, drink.

HDD (Honda Research Institute Driving Dataset) Ramanishka et al. (2018b) includes 137 driving videos with a total duration of 104 hours. Following prior work Wang et al. (2021b), we use the vehicle sensor as input signal and divide data into 100 sessions for training and 37 sessions for testing.

4.4.2 Implementation Details

For TVSeries and THUMOS'14, following Wang et al. (2021b); Xu et al. (2019); Eun et al. (2020); Gao et al. (2017a); Xu et al. (2021a), we resample the videos at 24 FPS (frames per second) and then extract frames at 4 FPS for training and evaluation. The sizes of *history* and *present* are set to 1024 and 8 most recently observed frames, respectively, spanning durations of 256s and 2s correspondingly at 4 FPS. For HDD, following OadTR Wang et al. (2021b), we extract the sensor data at 3 FPS for training and evaluation. The sizes of *history* and *present* are 48 and 6 most recently observed frames respectively, spanning durations of 16s and 2s correspondingly at 3 FPS.

Feature Extraction. Following Xu et al. (2021a); Wang et al. (2021b), we use mmaction2 Contributors (2020)-based two-stream TSN Wang et al. (2016) pretrained on Kinetics-400 Carreira and Zisserman (2017) to extract frame-level RGB and optical flow features for THUMOS'14 and TVSeries. We concatenate the RGB and optical flow features along channel dimension before feeding to the linear encoding layer in GateHUB. For HDD, we directly feed the sensor data as input to GateHUB. To fully leverage the proposed FaH, the feature extraction backbone needs to support multi-frame input. Since TSN only supports single-frame input, we explore spatiotemporal TimeSformer Bertasius et al. (2021) (pretrained on Kinetics-600 using 96×4 frame sampling) that supports multiple-frame input. We set the time duration for past t_{ps} and future t_f frames under FaH to be 1s and 2s respectively. We use TimeSformer to extract RGB features and use TSN-based optical flow features as TimeSformer only supports RGB. For our flow-free version, we replace optical flow features with features obtained from an additional multi-frame input of RGB frames uniformly sampled from a duration of 2s.

Training. We train GateHUB for 10 epochs using Adam optimizer Kingma and Ba (2015), weight decay of $5e^{-5}$, batch size of 50, OneCycleLR learning rate schedule of PyTorch Paszke et al. (2017) with pct_start of 0.25, $D = 1024$, latent encoding size $L = 16$, number of self-attention layers in History Decoder $N = 2$, $N_{heads} = 16$ for each attention layer and $\gamma_a = 0.6, \gamma_b = 0.2$ for background suppression.

Evaluation Metrics We follow the protocol of per-frame mean average precision (mAP) for THUMOS and HDD and calibrated average precision (mcAP) De Geest et al. (2016) for TVSeries.

4.4.3 Comparison with State-of-the-Art

Method	Feature Backbone		THUMOS14
	RGB	Optical Flow	mAP (%)
FATS Kim et al. (2021)	TSN	TSN	59.0
IDN Eun et al. (2020)			60.3
TRN Xu et al. (2019)			62.1
PKD Zhao et al. (2020)			64.5
OadTR Wang et al. (2021b)			65.2
WOAD Gao et al. (2021)			67.1
LSTR Xu et al. (2021a)			69.5
GateHUB (Ours)			**70.7**
TRN Xu et al. (2019)	TimeSformer	TSN	68.5
OadTR Wang et al. (2021b)			65.5
LSTR Xu et al. (2021a)			69.6
GateHUB (Ours)			**72.5**

Table 4.1 Online action detection results on THUMOS'14 comparing GateHUB with SoTA methods on mAP (%) when the RGB-based features are extracted from either TSN or TimeSformer. Optical flow-based features are extracted from TSN in all settings.

Table 4.1 compares GateHUB with existing state-of-the-art (SoTA) online action detection methods on THUMOS'14 for two different setups, one using RGB features from TSN Wang et al. (2016) and the other using RGB features from TimeSformer Bertasius et al. (2021). Both setups use optical flow features from TSN. WOAD Gao et al. (2021) uses RGB features from

I3D (equivalent to TSN). For TSN RGB features, all mAP in Table 4.1 are as reported in the references. For TimeSformer RGB features, we use the official code for TRN, OadTR and LSTR for fair comparison. From the table, we can observe that GateHUB outperforms all existing methods by at least 1.2% when using RGB features from TSN. Moreover, GateHUB outperforms existing methods by a larger margin of at least 2.9% using RGB features from TimeSformer. GateHUB is also the first approach to surpass 70% on THUMOS'14 benchmark. This validates that GateHUB, comprising GHU, Background Suppression and FaH to holistically leverage the long history more informatively, outperforms all SoTA on THUMOS'14.

We further compare GateHUB with SoTA on TVSeries and HDD in Table 4.2a and 4.2b, respectively. Following protocol, we use RGB and optical flow features from TSN for TVSeries and sensor data for HDD. All results from SoTA are as reported in the references. We can observe that GateHUB outperforms all SoTA on both TVSeries and HDD. The large improvement on HDD using sensor data validates that GateHUB is also effective on data modalities other than RGB or optical flow.

Method	mcAP (%)
FATS Kim et al. (2021)	84.6
IDN Eun et al. (2020)	86.1
TRN Xu et al. (2019)	86.2
PKD Zhao et al. (2020)	86.4
OadTR Wang et al. (2021b)	87.2
LSTR Xu et al. (2021a)	89.1
GateHUB (Ours)	**89.6**

(a)

Method	mAP (%)
CNN De Geest et al. (2016)	22.7
LSTM Ramanishka et al. (2018a)	23.8
RED Gao et al. (2017a)	27.4
TRN Xu et al. (2019)	29.2
OadTR Wang et al. (2021b)	29.8
GateHUB (Ours)	**32.1**

(b)

Table 4.2 Online action detection results comparing GateHUB with state-of-the-art methods on (a) TVSeries using RGB + Optical Flow data as input on mcAP metric and (b) HDD using sensor data as input on mAP metric.

4.4.4 GateHUB: Ablation Study

In this section, we conduct an ablation study to highlight the impacts of the novel components of GateHUB. Unless stated otherwise, all experiments are on THUMOS'14 using RGB and optical flow features from TSN.

Impact of Gated History Unit (GHU). We conduct an experiment where we test different variants of our Gated History Unit (GHU) by removing one or more of its design elements.

Method	mAP (%)
w/ GHU (Ours)	**70.7**
w/o GHU	69.6
w/ GHU suppress only	70.5
w/ GHU enhance only	70.5
w/ GHU w/o position-guidance	70.3
w/ GHU per head	68.0

(a)

Method	mAP (%)
Ours $\gamma_a > \gamma_b$	**70.7**
Ours $\gamma_a < \gamma_b$	70.2
w/ cross-entropy	69.9
w/ standard focal loss	70.2

(b)

Method	Future Duration	mAP (%)
w/o FaH	-	71.5
w/ FaH	0.5	71.1
	1s	72.0
	2s	**72.5**
	4s	71.4

(c)

Table 4.3 Ablation study comparing different variants of (a) Gated History Unit (GHU), (b) background suppression objective and (c) Future-augmented History (FAH). Ablation in (a) and (b) is conducted with RGB features from TSN and in (c) are conducted with RGB features from TimeSformer. Optical flow features from TSN are used in all settings.

Table 4.3a summarizes the results of this experiment. In the table, 'w/o GHU' refers to replacing GHU with vanilla cross-attention from Perceiver IO Jaegle et al. (2021a) and LSTR Xu et al. (2021a), *i.e.,* , CrossAttention(Q, K, V) = SoftMax$(QK^{\top}/\sqrt{d})$. In 'w/ GHU enhance only', we remove $\log(\mathbf{z}^{\mathbf{g}})$ from Eqn. 4.2 that suppresses history frames, *i.e.,* $G = \mathbf{z}^{\mathbf{g}}$. Conversely, in 'w/ GHU suppress only', we remove $\mathbf{z}^{\mathbf{g}}$ from Eqn. 4.2 that enhances history frames, *i.e.,* $G = \log(\mathbf{z}^{\mathbf{g}})$. In 'w/ GHU w/o position guidance', we operate on frame features before subjecting them to learned position encoding, *i.e.,* $G = \log(\mathbf{z}^{\tilde{\mathbf{g}}}) + \mathbf{z}^{\tilde{\mathbf{g}}}$ where $\mathbf{z}^{\tilde{\mathbf{g}}} = q(\mathbf{h})\mathbf{E}$. We also compare with 'w/ GHU per head' where G is learned separately for each cross-attention head.

Table 4.3a shows that our implementation of GHU significantly outperforms all other variants of GHU and cross-attention. We can observe that 'w/o GHU' performs 1.1% worse than 'w/ GHU'. This is because, without explicit gating, vanilla cross-attention fails to learn attention weights for history frames that correlate with how informative history frames are to current frame prediction (also depicted in Figure 4.1). Moreover, the lower performances of 'w/ GHU suppress only' and 'w/ GHU enhance only' validate that we need to both enhance the informative history frames and suppress the uninformative ones to achieve the best performance. Without the ability to both enhance and suppress, the model may encode uninformative history frames into the latent encoding or inadequately emphasize the informative ones, leading to worse performance. The performance is also lower when using history frame features without position encoding ('w/ GHU w/o position guidance'). This is because without position guidance, the model cannot assess the relative position of a particular history frame *w.r.t.* the current frame which is an important factor in deciding how informative a history frame is to current frame prediction. We also find having separate G per head (‘w/ GHU per head) performs much worse than sharing G across heads due to overfitting from N_{heads} times more parameters.

Impact of Background Suppression. We compare our background suppression objective with standard cross-entropy loss (*i.e.,* , $\gamma_a = \gamma_b = 0$) and standard focal loss(*i.e.,* , $\gamma_a = \gamma_b \neq 0$) Lin et al. (2017) as shown in Table 4.3b. First, compared to our background suppression objective, both standard cross-entropy and focal loss achieve lower accuracy. This validates that it is important to put separate emphasis on the low-confident action and background predictions to effectively differentiate action frames and closely resembling background frames. Furthermore, we find that across different combinations of γ_a and γ_b, choosing a pair where $\gamma_a > \gamma_b$ leads to higher accuracy. Specifically, we find $\gamma_a = 0.05$ and $\gamma_b = 0.025$ to give the highest accuracy. This can be attributed to the high data imbalance. Action frames are much lower in number than background frames and therefore require a stronger emphasis than the background.

Impact of Future-augmented History (FaH). Table 4.3c shows the ablation on FaH. Since the TSN backbone is not compatible with multi-frame input, we conduct this study using RGB features from TimeSformer. The table shows that with 2s of future information incorporated into history features, we achieve the best accuracy which is 1% higher than without future-augmented history (‘w/o FaH’). The accuracy is also improved with 1s of future information incorporated into history features. We further observe that the accuracy drops when future duration is much longer *e.g.,* 4s or much shorter *e.g.,* 0.5s. This shows that making a history frame aware of its future enables it to be more informative for current frame prediction. At the same time, future duration up to a certain extent (in our case, 2s) can encode meaningful future into history frames. Much beyond that, the future changes enough to be of little use for a given history frame, while much shorter future duration may also add noise rather than information. We wish to emphasize that all future duration are bound by the frames observed so far and do not extend into inaccessible future frames.

GateHUB Present Decoder. Table 4.4 shows the ablation study on our Present Decoder by

Method	mAP (%)
Ours	**70.7**
w/o self-attention	67.7
w/ cross-attention only at layer 1	68.6
w/ disjoint history and present	69.4

Table 4.4 Ablation study for Present Decoder by altering different aspects of the design.

altering different aspects of the design. Unlike the original PerceiverIO Jaegle et al. (2021a), where the output queries are independent, we model the present (equivalent of output queries in our method) via a causal self-attention and cross-attend it with history encoding multiple times (inspired by Perceiver Jaegle et al. (2021b)). We can observe in Table 4.4 that treating present frames independently (‘*i.e.,* w/o self-attention’) and having only one cross-attention (‘*i.e.,* w/ cross-attention only at first layer’) both reduce the accuracy considerably. Unlike LSTR Xu et al. (2021a) that uses a FIFO queue with disjoint long-term and short-term memory, in our design, the sequences of history and present frames fully overlap. Table 4.4 shows that having disjoint history and present frames (*i.e.,*, ‘w/ disjoint history and present’) leads to a 1.3% lower performance, further validating our design of Present Decoder and GateHUB overall.

4.4.5 GateHUB Efficiency

For online action detection setting, model efficiency is an important metric. We compare GateHUB with existing methods *w.r.t.* parameter count, GFLOPs, and inference speed in terms of FPS as shown in Table 4.5. We first observe that GateHUB achieves the highest accuracy with the least number of model parameters compared to all existing methods. We also note that while methods like OadTR Wang et al. (2021b) and TRN Xu et al. (2019) are more efficient in terms of GFLOPs, their accuracy is much lower. GateHUB achieve a more favorable accuracy-efficiency trade-off with fewer GFLOPs than the existing best method LSTR Xu et al. (2021a) while obtaining a higher accuracy. All aforementioned methods require optical flow computation which is time-consuming, therefore the inference speed of these methods is governed by the optical flow computation speed of 8.1 FPS. Meanwhile, our flow-free model obviates optical flow computation by using RGB features from TimeSformer at two different frame rates and attains higher or close accuracy compared to existing work at 2.8× faster inference speed. When compared with flow-free

Method	Model		Inference Speed (FPS)					mAP(%)
	Parameter Count	GFLOPs	Optical Flow Computation	RGB Feature Extraction	Flow Feature Extraction	Model	Overall	
TRN Xu et al. (2021b)	402.9M	1.46	8.1	70.5	14.6	123.3	8.1	62.1
OadTR Wang et al. (2021b)	75.8M	2.54	8.1	70.5	14.6	110.0	8.1	65.2
LSTR Xu et al. (2021a)(Flow-free)	54.2M	6.33	-	22.7	-	99.2	22.7	63.5
LSTR Xu et al. (2021a)	58.0M	7.53	8.1	70.5	14.6	91.6	8.1	69.5
Ours (Flow-free)	41.8M	3.47	-	22.7	-	83.3	**22.7**	66.5
Ours	45.2M	6.98	8.1	70.5	14.6	71.2	8.1	**70.7**

Table 4.5 Efficiency comparison of GateHUB using RGB and optical flow features and our optical flow-free version with existing methods. GateHUB using RGB and optical flow has the least parameter count compared to existing methods, and higher accuracy and lower GFLOPs than the existing best method. Moreover, our flow-free version attains higher or close accuracy compared to existing methods that require RGB and optical flow features at 2.8× faster inference speed.

LSTR, GateHUB achieves 3% higher mAP, thus providing a significantly better speed-accuracy tradeoff than the existing best method.

4.4.6 Qualitative Evaluation

Gated History Unit (GHU). We qualitatively assess the effect of GHU by visualizing examples of the most suppressed and most enhanced history frames in a streaming video when ordered as per the gating scores G learned by GHU in Eqn. 4.2. Fig. 4.3 shows examples from three videos where frames in the same row belong to the same video. From the figure, we can observe that GHU learns to suppress frames that exhibit no discernible action from the C action classes. The suppressed frames either have people arbitrarily moving or are uninformative background frames (*e.g.,* crowd cheering) that convey no useful information to predict action for the current frame. On the other hand, GHU learns to maximize emphasis on history frames with action from the C classes and on background frames that provide meaningful context to determine the current frame action (*e.g.,* long jump athlete running toward the pit).

Current Frame Prediction. We visualize GateHUB's current frame prediction in Fig. 4.4. The confidence in the range $[0, 1]$ on y-axis denotes the probability of predicting the correct action (*i.e., High Jump* in Fig. 4.4). We can observe that GateHUB with GHU (red) is effective in reducing false positives for background frames that closely resemble action frames compared to without GHU (orange).

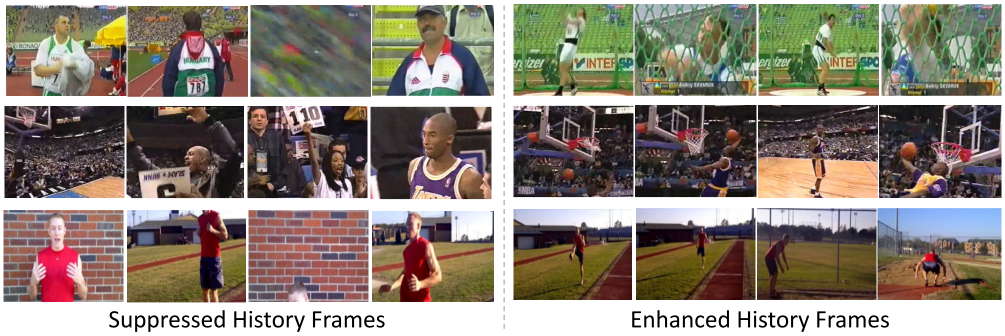


Figure 4.3 Examples of the most suppressed and most enhanced history frames as per the gating score learned by GHU. Frames in the same row belong to the same video.

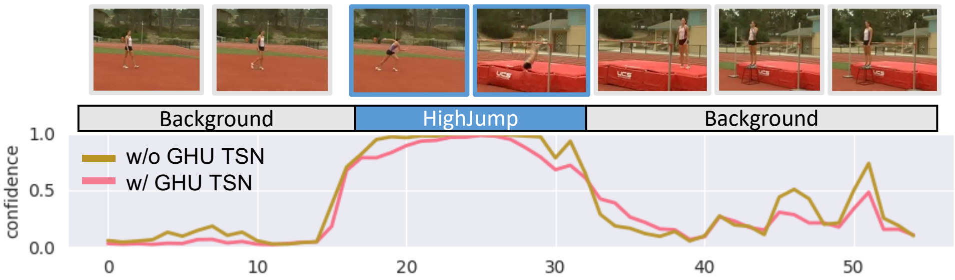


Figure 4.4 Visualization of GateHUB's online prediction. The curves indicate the predicted confidence of the ground-truth class (*High Jump*) using TSN backbone with and without GHU.

4.5 Summary

We present GateHUB for online action detection in untrimmed streaming videos. It consists of novel designs including Gated History Unit (GHU), Future-augmented History (FaH), and a background suppression loss to more informatively leverage history and reduce false positives for current frame prediction. GateHUB achieves higher accuracy than all existing methods for online action detection, and is more efficient than the existing best method. Moreover, its optical flow-free variant is 2.8× faster than previous methods that require both RGB and optical flow while obtaining higher or close accuracy.

While GateHUB outperforms all existing methods, there is ample room for improvement. Although GateHUB can leverage long history, the length is still finite and may not be adequate when actions occur infrequently over long duration. It would be worthwhile to investigate ways to leverage history sequences of any length. Another challenge is slow motion action which is uncommon and can have considerably different temporal distribution, making it difficult to predict as accurately as common actions.

BIBLIOGRAPHY

Arnab, A., Dehghani, M., Heigold, G., Sun, C., Lučić, M., and Schmid, C. (2021). Vivit: A video vision transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 6836–6846.

Beltagy, I., Peters, M. E., and Cohan, A. (2020). Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*.

Bertasius, G., Wang, H., and Torresani, L. (2021). Is space-time attention all you need for video understanding?

Buch, S., Escorcia, V., Shen, C., Ghanem, B., and Niebles, J. C. (2017). SST: Single-stream temporal action proposals. In *CVPR*.

Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., and Zagoruyko, S. (2020). End-to-end object detection with transformers. In *ECCV*.

Carreira, J. and Zisserman, A. (2017). Quo vadis, action recognition? A new model and the Kinetics dataset. In *CVPR*.

Chen, M., Radford, A., Child, R., Wu, J., Jun, H., Luan, D., and Sutskever, I. (2020). Generative pretraining from pixels. In *ICML*.

Chen, X., Wu, Y., Wang, Z., Liu, S., and Li, J. (2021). Developing real-time streaming transformer transducer for speech recognition on large-scale dataset. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*.

Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y. (2014). Learning phrase representations using rnn encoder-decoder for statistical machine translation.

Contributors, M. (2020). Openmmlab's next generation video understanding toolbox and benchmark. https://github.com/open-mmlab/mmaction2.

Dai, Z., Yang, Z., Yang, Y., Carbonell, J., Le, Q. V., and Salakhutdinov, R. (2019). Transformer-XL: Attentive language models beyond a fixed-length context.

De Geest, R., Gavves, E., Ghodrati, A., Li, Z., Snoek, C., and Tuytelaars, T. (2016). Online action detection. In *ECCV*.

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

Eun, H., Moon, J., Park, J., Jung, C., and Kim, C. (2020). Learning to discriminate information for online action detection. In *CVPR*.

Eun, H., Moon, J., Park, J., Jung, C., and Kim, C. (2021). Temporal filtering networks for online action detection. *Pattern Recognition*.

Fan, H., Xiong, B., Mangalam, K., Li, Y., Yan, Z., Malik, J., and Feichtenhofer, C. (2021). Multiscale vision transformers.

Gao, J., Yang, Z., and Nevatia, R. (2017a). RED: Reinforced encoder-decoder networks for action anticipation. In *BMVC*.

Gao, J., Yang, Z., Sun, C., Chen, K., and Nevatia, R. (2017b). TURN TAP: Temporal unit regression network for temporal action proposals. In *ICCV*.

Gao, M., Zhou, Y., Xu, R., Socher, R., and Xiong, C. (2021). WOAD: Weakly supervised online action detection in untrimmed videos. In *CVPR*.

Girdhar, R. and Grauman, K. (2021). Anticipative video transformer.

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

Idrees, H., Zamir, A. R., Jiang, Y.-G., Gorban, A., Laptev, I., Sukthankar, R., and Shah, M. (2017). The THUMOS challenge on action recognition for videos "in the wild". *CVIU*.

Jaegle, A., Borgeaud, S., Alayrac, J.-B., Doersch, C., Ionescu, C., Ding, D., Koppula, S., Zoran, D., Brock, A., Shelhamer, E., et al. (2021a). Perceiver io: A general architecture for structured inputs & outputs. *arXiv preprint arXiv:2107.14795*.

Jaegle, A., Gimeno, F., Brock, A., Zisserman, A., Vinyals, O., and Carreira, J. (2021b). Perceiver: General perception with iterative attention. *arXiv:2103.03206*.

Kim, Y. H., Nam, S., and Kim, S. J. (2021). Temporally smooth online action detection using cycle-consistent future anticipation. *Pattern Recognition*.

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

Lin, T., Liu, X., Li, X., Ding, E., and Wen, S. (2019). BMN: Boundary-matching network for temporal action proposal generation. In *ICCV*.

Lin, T., Zhao, X., Su, H., Wang, C., and Yang, M. (2018). BSN: Boundary sensitive network for temporal action proposal generation. In *ECCV*.

Lin, T.-Y., Goyal, P., Girshick, R., He, K., and Dollár, P. (2017). Focal loss for dense object detection. In *ICCV*, pages 2980–2988.

Liu, X., He, P., Chen, W., and Gao, J. (2019). Multi-task deep neural networks for natural language understanding. *arXiv preprint arXiv:1901.11504*.

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

Nawhal, M. and Mori, G. (2021). Activity graph transformer for temporal action localization. *arXiv:2101.08540*.

Neimark, D., Bar, O., Zohar, M., and Asselmann, D. (2021). Video transformer network. *arXiv:2102.00719*.

Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., and Lerer, A. (2017). Automatic differentiation in pytorch.

Qu, S., Chen, G., Xu, D., Dong, J., Lu, F., and Knoll, A. (2020). LAP-Net: Adaptive features sampling via learning action progression for online action detection. *arXiv:2011.07915*.

Ramanishka, V., Chen, Y.-T., Misu, T., and Saenko, K. (2018a). Toward driving scene understanding: A dataset for learning driver behavior and causal reasoning. In *CVPR*, pages 7699–7707.

Ramanishka, V., Chen, Y.-T., Misu, T., and Saenko, K. (2018b). Toward driving scene understanding: A dataset for learning driver behavior and causal reasoning. In *CVPR*.

Shou, Z., Chan, J., Zareian, A., Miyazawa, K., and Chang, S.-F. (2017). CDC: Convolutional-de-convolutional networks for precise temporal action localization in untrimmed videos. In *CVPR*.

Shou, Z., Wang, D., and Chang, S.-F. (2016). Temporal action localization in untrimmed videos via multi-stage cnns. In *CVPR*.

Tan, J., Tang, J., Wang, L., and Wu, G. (2021). Relaxed transformer decoders for direct action proposal generation. *arXiv:2102.01894*.

Touvron, H., Cord, M., Douze, M., Massa, F., Sablayrolles, A., and Jégou, H. (2021). Training data-efficient image transformers & distillation through attention. In *ICML*.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. In *Advances in neural information processing systems (NeurIPS)*, volume 30.

Wang, L., Xiong, Y., Wang, Z., Qiao, Y., Lin, D., Tang, X., and Van Gool, L. (2016). Temporal segment networks: Towards good practices for deep action recognition. In *ECCV*.

Wang, L., Yang, H., Wu, W., Yao, H., and Huang, H. (2021a). Temporal action proposal generation with transformers. *arXiv preprint arXiv:2105.12043*.

Wang, S., Li, B., Khabsa, M., Fang, H., and Ma, H. (2020). Linformer: Self-attention with linear complexity. *arXiv:2006.04768*.

Wang, X., Zhang, S., Qing, Z., Shao, Y., Zuo, Z., Gao, C., and Sang, N. (2021b). Oadtr: Online action detection with transformers.

Xu, H., Das, A., and Saenko, K. (2017). R-C3D: Region convolutional 3d network for temporal activity detection. In *ICCV*.

Xu, M., Gao, M., Chen, Y.-T., Davis, L. S., and Crandall, D. J. (2019). Temporal recurrent networks for online action detection. In *ICCV*.

Xu, M., Xiong, Y., Chen, H., Li, X., Xia, W., Tu, Z., and Soatto, S. (2021a). Long short-term transformer for online action detection.

Xu, W., Xu, Y., Chang, T., and Tu, Z. (2021b). Co-scale conv-attentional image transformers. *arXiv:2104.06399*.

Zhang, C., Gupta, A., and Zisserman, A. (2021). Temporal query networks for fine-grained video understanding. In *CVPR*, pages 4486–4496.

Zhao, P., Wang, J., Xie, L., Zhang, Y., Wang, Y., and Tian, Q. (2020). Privileged knowledge distillation for online action detection. *arXiv:2011.09158*.

Zhao, Y., Xiong, Y., Wang, L., Wu, Z., Tang, X., and Lin, D. (2017). Temporal action detection with structured segment networks. In *ICCV*.

Zhu, Z., Tang, W., Wang, L., Zheng, N., and Hua, G. (2021). Enriching local and global contexts for temporal action localization. In *ICCV*, pages 13516–13525.

CHAPTER 5

ACTIVITY-DRIVEN WEAKLY-SUPERVISED SPATIAL-TEMPORAL VIDEO OBJECT GROUNDING

5.1 Introduction

Grounding natural language in visual data is a fundamental task in the multimedia and computer vision communities with a variety of applications, including image/video retrieval Karpathy and Fei-Fei (2015), robotics Alomari et al. (2017) and human-computer interactions Shridhar and Hsu (2018). Given an image/video and its description sentence, for example "break the eggs", visual grounding aims at localizing the query objects described in the sentence on the given image or video. Recently, great progress has been made on image grounding Yang et al. (2019b); Karpathy and Fei-Fei (2015); Chen et al. (2018); Yang et al. (2019a). On the basis of this, researchers started to explore grounding in the video domain Zhou et al. (2018a); Shi et al. (2019); Chen et al. (2019b,a); Huang et al. (2018).

Nevertheless, in video grounding, it is labor-intensive to annotate a considerable number of bounding boxes for queries in videos. To address this challenge, multiple instance learning (MIL) methods Zhou et al. (2018a); Shi et al. (2019); Chen et al. (2019b,a); Huang et al. (2018) were proposed, which do not require bounding box annotations in the training videos. Video object grounding is achieved in a weakly-supervised fashion, where only a video and its description sentence are required during training. However, these methods are only able to infer the spatial occurrence of the query objects, and cannot tell the temporal occurrence of the objects. This problem was later addressed in Chen et al. (2019b) by generating region proposal tubes using object tracking methods. But their method is only applicable to trimmed videos without camera shot cuts.

We argue that a successful video grounding method should infer both the spatial and temporal occurrence of a query object without the need of expensive annotations. In addition, the method is expected to work in untrimmed videos, which can be of long duration and contain frequent visual inconsistency mainly caused by frame flickerings and camera shot cuts (see Fig. 5.1). A

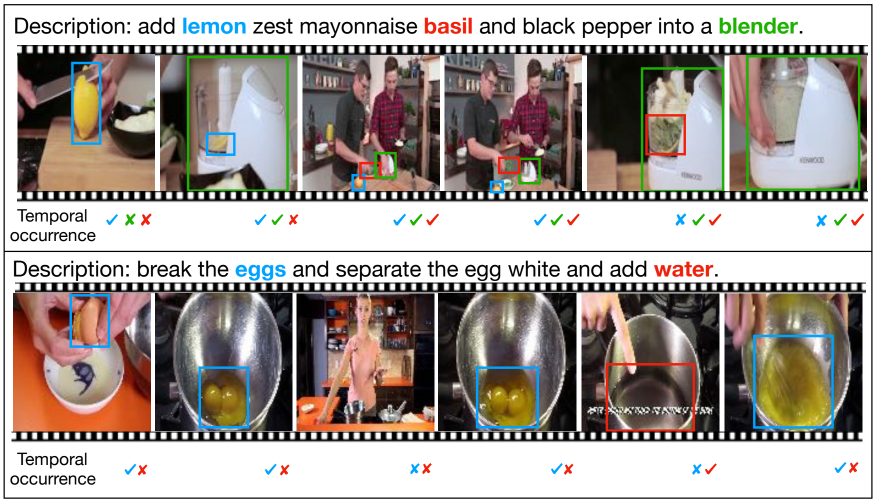

Description: add lemon zest mayonnaise basil and black pepper into a blender.
Temporal occurrence
✓✗✗
✓✓✗
✓✓✓
✓✓✓
✗✓✓
✗✓✓
Description: break the eggs and separate the egg white and add water.
Temporal occurrence
✓✗
✓✗
✗✗
✓✗
✗✓
✓✗

Figure 5.1 Given a video and its description sentence, our goal is to achieve the spatio-temporal grounding of the described queries on challenging untrimmed videos, where camera shot cuts frequently appear. Spatio-temporal grounding grounds each query to specific spatial regions and the frames of a video where the query object appears.

query object may appear discontinuously (it frequently appears and disappears) across frames in an untrimmed video. Existing video grounding methods Chen et al. (2019b) that rely on visual trackers Wang et al. (2019b,a) would undoubtedly fail as the trackers can be distracted if a camera shot cut appears.

We propose a novel multiple instance learning method for spatio-temporal grounding on untrimmed videos. Our method does not require extensive annotations of spatial and temporal occurrence[1] of the query object in training. At the spatial level, we assign each textual query to one of the region proposals in a frame, while at the temporal level, we represent each frame by query-specific region and ground each query to its relevant frames. We formulate spatio-temporal grounding as two MIL problems. The spatial MIL aims at selecting the best instance (top-ranked region) from a bag (frame). The temporal MIL aims at selecting the multiple instances (query occurring frames) from a bag (video). Two MILs are mutually guided to achieve the optimal spatio-temporal grounding results.

We also propose to model human activity operating on the query object. This allows us to capture the physical states of the object as well as the spatial relations between human and the

[1] Temporal occurrence of an object means the object appear in some frames of a video.

object. Most of existing visual grounding methods Karpathy and Fei-Fei (2015); Shi et al. (2019); Zhou et al. (2018a); Chen et al. (2019a) simply compute the similarity between the visual and the textual features of the query object as a measurement for selecting candidate regions for the query. However, there is a granularity gap between the coarse textual and rich visual modalities. For example, the text-level query object “potatoes” might correspond to “potatoes” in different physical states: “mashed potatoes” means visually paste-like potatoes, while “peel potatoes and cut” corresponds to cube-shape potatoes. Directly computing the feature similarity as in Karpathy and Fei-Fei (2015); Shi et al. (2019); Zhou et al. (2018a) leads to a large discrepancy between the text and visual features. To address this, first, we propose to enrich the textual representation by incorporating the activity performed on the query to better align the text feature with the diverse visual features. Second, we propose an activity-driven region proposal refinement to find high-quality region proposals. Most of existing visual grounding methods Shi et al. (2019); Zhou et al. (2018a); Chen et al. (2019a); Karpathy and Fei-Fei (2015) build a candidate pool of top-N region proposals, in which query objects could be missed. Proposals with a high recall rate by increasing N typically lead to a large search space for grounding a query object. To tackle this dilemma, we exploit the intrinsic spatial relations between human and object using human activity to refine the search space for region proposal generation.

Our work is different from Zhou et al. (2018a); Shi et al. (2019), which also focus on grounding untrimmed videos. However, they ground query objects every frame even if the frame does not contain the query. This could result in a lot of false positives in the frames where the queried object does not appear due to its sparse existence in a long untrimmed video. On the contrary, we infer both the bounding box and the temporal occurrence of the query object. Therefore, our method can be used in more realistic scenarios.

Our main contribution can be summarized as follows: 1). We propose a spatio-temporal Multiple Instance Learning method to learn a spatio-temporal video grounding model for the challenging untrimmed videos in a weakly-supervised fashion; 2). We exploit the activity cues in the description sentence of the video, including enriching the query representation with activity effect and refine the object proposal generation; 3). Extensive results demonstrate that our method outperforms state-of-the-art weakly-supervised object grounding model in untrimmed videos by a large margin.

5.2 Related Work

Weakly-supervised Visual Grounding. Weakly-supervised image grounding Karpathy and Fei-Fei (2015); Chen et al. (2018); Rohrbach et al. (2016) has been extended to video domain Zhou et al. (2018a); Shi et al. (2019); Huang et al. (2018); Chen et al. (2019b,a), but they are only applicable to constrained scenarios. Early work Yu and Siskind (2013) grounded sentences to objects in the constrained videos that are recorded in lab. A reference grounding model Huang et al. (2018) extends proposal ranking Karpathy and Fei-Fei (2015) to video domain and further enhances the performance by modeling the reference relationships between video segments. Following Karpathy and Fei-Fei (2015), the work in Zhou et al. (2018a) extends proposal ranking to video domain via a frame-wise weighting strategy. They also introduce an object grounding dataset based on YouCookII Zhou et al. (2018b). The work in Shi et al. (2019); Chen et al. (2019a) follow the same problem setup as Zhou et al. (2018a) and boost grounding performance by using contextual similarity and cross-modal context reasoning. However, during inference Shi et al. (2019); Zhou et al. (2018a); Chen et al. (2019a) only ground query in the frames where the objects occur without grounding frame occurrence in the temporal domain. Thus, the output of their methods contain a lot of false positives in the frames without the presence of query objects. The VID-sentence dataset is introduced in Chen et al. (2019b), which first grounds spatio-temporal tubes for a query. But their method and dataset are only for trimmed videos.

In this work, we aim at object grounding on untrimmed video streams by localizing the query objects in both spatial region and frame-level occurrence. Our method does not rely on tracking tubes due to frame flickering and camera shot cut in untrimmed videos.

Fully-supervised Spatio-Temporal Grounding has been developed by combining with other tasks, such as object tracking Yang et al. (2019c), video captioning Zhou et al. (2019) and visual question answering Cadene et al. (2019). Yang et al. (2019c) add language description on an

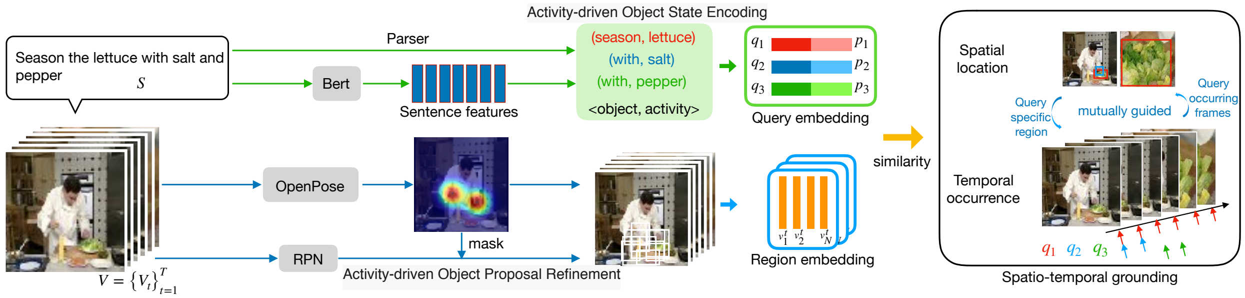


Figure 5.2 Overview of our framework. Given a video and its description sentence as input, we first extract the region features via a pre-trained region proposal network (RPN) and find the high-quality proposals by the proposed activity-driven object proposal refinement module. Given text data, we first encode the sentence by BERT and propose an activity-driven object state encoding module to enrich query representation by incorporating activity effect. Then, through a similarity alignment between the two modalities, spatio-temporal grounding is achieved by retrieving the frames that contain the queries and selecting the best-match spatial region in the frames where the queries appear. During training, the spatial and temporal levels are mutually guided. Training details can be seen in Section 5.3.3.

object tracking dataset Fan et al. (2019) to make it for grounding task and propose a grounding and tracking integration model. But this dataset only contains single object in a video, which is much easier than our goal that multiple objects in a query need to be grounded. Zhou et al. (2019) augment the challenging ActivityNet Captions dataset with 158K bounding boxes annotations and provide a framework to not only generate video captions but also link the sentence to the evidence in the video. However, these methods require dense spatio-temporal tube annotations for training, which are especially expensive to obtain. Our paper aims at solving spatio-temporal grounding in a weakly-supervised setting without bounding box annotations.

Weakly-supervised Video Object Localization localizes an object class or a video tag in the visual content. Object class or video tag comes from human labeling while the descriptive sentence in visual grounding can be accessed from the existing web video descriptions uploaded by users or the YouTube Automatic Speech Recognition scripts, which requires less human effort. Existing work of weakly-supervised object localization also formulate it as an MIL approach. Kwak et al. (2015) integrated object tracking and frame-wise object detection together to achieve video object localization. Prest et al. (2012) extract spatio-temporal tubes as proposals to be ranked and selected. However, similar to Yang et al. (2019c); Chen et al. (2019b), these methods heavily rely on tracking which is not applicable to long untrimmed video due to camera shot cut.

Weakly-supervised Temporal Grounding focuses on identifying relevant frames in a video from text descriptions without the annotations of temporal boundaries. Existing work Mithun et al. (2019); Gao et al. (2019); Duan et al. (2018); Chen et al. (2020) extract a set of pre-defined temporal segment proposals and select one of them that semantically best matches the description. Our task is more challenging as we ground the query not only to its occurring frames but also to specific locations. Moreover, query appears discontinuously in temporal domain due to the frequent camera shot cut, which may not be addressed by finding the best matched temporal proposal.

5.3 Our Approach

5.3.1 Problem Setup

Given an untrimmed video and its description sentence, video grounding task grounds each query object described in the sentence to spatio-temporal visual regions in the video. The query can be either a noun, e.g. word "potato" or a pronoun with reference meaning, e.g. "they". Video grounding on untrimmed videos is of great significance while more challenging than trimmed video, since the untrimmed video contains large temporal incoherence caused by camera motion and camera shot cut[2].

We propose a spatio-temporal grounding model that can be applied to untrimmed videos with significant camera motion. Our model is trained in a weakly-supervised fashion, where only the video-description pairs are given during training; spatial and temporal occurrence of the query objects are not given. As shown in Fig. 5.2, our grounding model takes a video V and its description sentence S as a pairwise input, and predicts the temporal occurrence (*i.e.,* , what frames contain the object) of the query object across frames and the spatial location (using a bounding box) of the object on the frames where it appears. The video contains T frames and is denoted as $V = \{V_t\}_{t=1}^{T}$. Following Shi et al. (2019); Zhou et al. (2018a), each frame consists of N region proposals, denoted as $V_t = \left\{v_t^n\right\}_{n=1}^{N}$, where n indexes the proposals in the t-th frame. The description sentence S includes K queries, e.g. query "lettuce" and "pepper" in the description sentence "season the

[2] A camera shot cut is the view change from one shot to another, e.g., from a distant view shot to a close view shot.

lettuce with salt and pepper". Each query s_k corresponds to a word or a phrase in S and all of queries in a sentence is denoted as $\{s_k\}_{k=1}^{K}$. The visual feature v_t^n and query feature Q_k of a query are encoded into a joint feature space, and their similarity is computed for the grounding purpose.

We formulate spatio-temporal grounding as a multiple instance learning (MIL) problem for untrimmed videos. We propose two ranking losses (one on spatial level and one on temporal level) mutually guiding each other to learn a shared metric space for grounding. We consider a weakly-supervised learning scenario without the annotations of bounding boxes and temporal occurrences. An activity-driven encoder is proposed to better align the visual and text modalities by considering the object state variations and spatial location prior of region proposals.

5.3.2 Activity-driven Encoding

Activity cues in both text and visual modalities are informative for grounding objects in an untrimmed video. For example, as shown in Fig. 5.3, the activity in the description sentence "mash the potatoes" results in paste-like potatoes in the visual data, while "peel potatoes and cut" results in cube-shaped potatoes. By modeling activities, various physical states of an object can be modeled at a fine-grained representation level, which allows us to accurately ground the object. In addition, the activity provides a spatial location prior for the object to be grounded. For example, "cut potatoes" indicates that query potatoes should appear close to human hand. The spatial location prior can be exploited to refine the candidate region proposals of visual data.

5.3.2.1 Activity-driven Object State Encoding

To encode the query into a representative feature, previous work Shi et al. (2019); Zhou et al. (2018a) extract each query word (e.g. "potatoes") from the description sentence and then represent it based on GloVe features Pennington et al. (2014). This is ineffective because there is a semantic granularity gap between the text modality and visual modality (see Fig. 5.3). Existing methods simply attach the same textual representation to the diverse visual representation, which results in text-visual misalignment problem.

We propose to enrich the textual representation and align it with diverse visual objects. This allows us to capture rich cues of object physical states. Specifically, we introduce an activity-driven

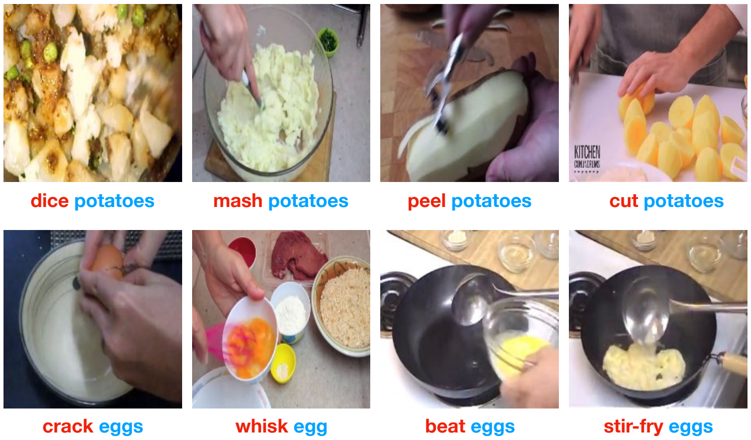


Figure 5.3 Examples of an object in different states. For example, "potatoes" in "mashed" and "peeled" are different in appearance. Blue word indicates query object and red word indicate the activity applied on the object. Best view in color.

object state encoding module to enrich the query textual representation. We consider each query with the predicate performing on it and reformulate each query as an object-activity pair. We use Stanford CoreNLP parser Manning et al. (2014) to parse each noun or pronoun and its predicate from a sentence S. Meanwhile, each sentence is encoded by a pre-trained BERT model Devlin et al. (2019). Then, we crop the features of the k-th query and its predicate from the sentence representation as q_k and p_k, respectively. The textual representation of k-th query s_k in the sentence is enriched as (q_k, p_k), which is an object-activity pair. Query set $Q = \{(q_1, p_1), ..., (q_K, p_K)\}$ is denoted as the textual representations of every query in sentence. With activity-driven object states encoding, textual and visual modalities can be better aligned without the large granularity gap.

5.3.2.2 Activity-driven Object Proposal Refinement

Most of existing visual grounding methods Karpathy and Fei-Fei (2015); Shi et al. (2019); Zhou et al. (2018a); Chen et al. (2019a) are based on selecting the best matched proposal out of a candidate pool that contains top-N region proposals as a grounded object. However, query objects may not be included in these proposals and thus are unlikely to be grounded. A naive solution is to increase N but this will lead to a large search space especially for long untrimmed videos.

Human activity can provide spatial location prior to refine the region proposal generation. Intuitively, there is a spatial dependency between the activity performer and the activity receiver. For example, if a video is describing "peel a potato", "potato" tends to occur around "human" hands, which indicates the potential of spatial location prior between activity performer (human) and receivers (objects). We propose to model the spatial prior as a truncated normal distribution $\mathcal{N}(\mu_a, \nu_a)$ to mask out the irrelevant region proposals. μ_a is the pixel coordinate of activity-relevant joint and ν_a is the hyperparameter. We apply the normal distribution to activity-relevant key joints to form a mixture of Gaussian (see the heatmap in Fig. 5.2), and select top-N proposals by considering the densities. We use a pre-trained human detector He et al. (2017) and OpenPose Cao et al. (2019) to extract the human's joints. If there is no human detected in a frame like Fig. 5.3, most likely the frame is captured in a close view. Then, we simply keep the top-N region proposals without refinement. Using this method, we can include more query-related proposals and make the query more likely to be grounded.

5.3.3 Spatio-Temporal MIL for Video Grounding

We consider a weakly-supervised learning scenario, where the only supervision is the sentence description of the video. Existing work Chen et al. (2019b) addresses the weakly-supervised spatio-temporal grounding for trimmed videos using object tracking to generate proposal tubes. Different from Chen et al. (2019b), our goal is to achieve *untrimmed* video grounding. This is more challenging as an object does not necessarily appear on every frame and usually occur discontinuously due to the frequent camera shot cut in untrimmed videos. In this case, the tracking-based grounding methods would undoubtedly fail.

To address this problem, we resort to MIL and propose a novel spatio-temporal MIL framework. At the spatial level, we aim at grounding each textual query to one of N object proposals v_t^n ($n \in [1, N]$) extracted on a frame. At the temporal level, we aim at grounding each textual query to the frames where it occurs on a video. Both the temporal and spatial MILs are formulated by pair-wise ranking losses. The losses encourage the correct matching of an aligned video-sentence pair and discourages the matching of an unaligned pair, *i.e.,* , the sentence does not belong to the video[3]. The spatial and the temporal MILs are mutually guided to learn a spatio-temporal

[3] We consider a video and its descriptive sentence as an aligned video-sentence pair and define a video and the query in its descriptive sentence as an aligned video-query pair. Similarly, an unaligned video-sentence pair is that the sentence does not describe the video but describes other video in the current batch.

grounding model.

5.3.3.1 Spatial level MIL

The goal of spatial grounding is to ground each referred query to one of the top-N region proposals on a frame. In order to obtain the query specific region, we use normalized cosine distance as the metric of region-query similarity $a(v_t^n, Q_k)$:

$$a(v_t^n, Q_k) = \frac{v_t^{n\mathrm{T}} \cdot (q_k + p_k)}{\left\|v_t^n\right\| \|(q_k + p_k)\|}, \tag{5.1}$$

where $(q_k + p_k)$ and v_t^n are feature embeddings of the textual object-activity pair of k-th query and the region proposal, respectively, in a joint d-dimensional feature space. k, t, n index queries, frame, and region proposals, respectively. T is a transpose.

The spatial-level MIL regards each frame as a bag and all region proposals in the frame as instances in the bag. Instance score *w.r.t.* query Q_k is the region-query similarity $a(v_t^n, Q_k)$ computed by Eq. 6.4. Following MIL, a bag is represented by its most positive instance, which can be achieved by a *max* operation. Thus, the bag level score is computed as $S(V_t, Q_k) = \underset{n}{max}\, a(v_t^n, Q_k)$, which denotes the frame-query similarity. Following Shi et al. (2019); Chen et al. (2019b); Zhou et al. (2018a), spatial MIL is formulated as a pair-wise ranking:

$$S(V_t, Q_k) > max\left(S(V_t', Q_k), S(V_t, Q_j')\right), \tag{5.2}$$

where (V_t, Q_j') and (V_t', Q_k) are the two cases of unaligned frame-query pairs. Q_j' is the j-th unaligned query w.r.t. region proposal V_t, while V_t' consists of region proposals in a video frame unaligned with current query Q_k. Eq. 5.2 encourages the correct proposal matching for a query Q_k by $S(V_t, Q_k) > S(V_t', Q_k)$ and encourages the correct query matching for a frame V_t by $S(V_t, Q_k) > S(V_t, Q_k')$.

To achieve the pair-wise ranking in Eq. 5.2, the frame-query ranking loss with margin Δ_s needs to be minimized in training:

$$\mathcal{L}_{rank}^{t,(k,j)} = max\left(0, max\left(S(V_t', Q_k), S(V_t, Q_j')\right) - S(V_t, Q_k) + \Delta_s\right). \tag{5.3}$$

This objective encourages the similarities of aligned pairs larger than those of unaligned pair with gap Δ_s. Furthermore, by aggregating every query in the unaligned description sentence, the spatial-level ranking loss is defined as:

$$\mathcal{L}_{rank}^{t,k} = \frac{1}{K'} \sum_{j=1}^{K'} \mathcal{I}(Q'_j \neq Q_k) \mathcal{L}_{rank}^{t,(k,j)}, \tag{5.4}$$

where the negative query set Q' contains K' queries. Note that if Q_k meets the query Q'_j in negative query set Q', it will not contribute to the ranking loss. But if the queries in Q and Q' only share the object and have different activities, such as "mash the potato" and "peel the potato", it still contributes to the ranking loss, because of the large discrepancy in appearance.

Spatial MIL considers each frame as a bag. However, in untrimmed videos, query only appears in a part of frames. The frames without the query occurring are actually noisy positive bags. In Sec 3.3.2 and 3.3.3, we will discuss how to alleviate the false positive bags with the guidance of temporal grounding.

5.3.3.2 Temporal level MIL

In temporal grounding, we aim at predicting the temporal occurrence of the queries across frames. In our weakly-supervised setting, we do not have access to the temporal occurrence annotations. Thus, we still formulate it as a MIL problem. In this case, each video is considered as a bag and the frames of the video are considered as instances in the bag.

Instance score is frame-query similarity. But query object only occurs in a small region of the frame. It is not effective to align the query with the entire frame. Thus, we resort to spatial level MIL results as guidance and propose to represent each instance as the best matched region proposal such that the instance score is denoted as $S(V_t, Q_k) = \underset{n}{max}\, a(v_t^n, Q_k)$.

In an untrimmed video, the query object may appear discontinuously across frames. Thus, it is not appropriate to represent the bag by the best-matched instance, as it ignores other positive instances that contain the query. This is different from the spatial level MIL where an object tends to appear concentrated in frame. Thus, in temporal level MIL, the bag score which is video-query pair-wise similarity should be the overall score of all positive instances in the bag $\frac{1}{T} \sum_{t \in T} S(V_t, Q_k)$, instead of the best matched instance. Since we have video-query pair as bag-level annotation, temporal level MIL is formulated as a ranking problem:

$$\frac{1}{T}\sum_{t=1}^{T} S(V_t, Q_k) > max\left(\frac{1}{T}\sum_{t=1}^{T} S(V'_t, Q_k), \frac{1}{T}\sum_{t=1}^{T} S(V_t, Q'_j)\right), \tag{5.5}$$

which is also a pair-wise ranking. V'_t and Q'_j indicate the negative video frame and the j-th query in the negative query set, respectively. Eq. 5.5 encourages an aligned video-query pair (V, Q_k) to be better matched than two other types of unaligned video-query pairs (V', Q_k) and (V, Q'_j). The number of instances in each bag is T. Using average instance scores to represent a bag score helps avoid a degenerate solution where we predict most of frames irrelevant to the queries, compared with *max* operation.

Moreover, consecutive frames in a video are correlated but their visual context does not necessarily to be continuous due to the frequent camera shot cut. Therefore, the simple arithmetic mean in Eq. 5.5 is not adaptive for temporal grounding in untrimmed videos. In this paper, we propose an attention module to learn the the weight of each frame. Specifically, we extract the each frame's features from the last layer of VGG-16 backbone and then encode the frames' features by a self-attention layer as $f_t, t \in [1, T]$. Based on that, we compute the weight of each frame as w_t by a linear layer and a sigmoid activation w.r.t f_t. Note that the weight of each frame is agnostic to different queries while the video content continuity can be addressed by the temporal consistency of frame weights.

To achieve the goal of Eq. 5.5 by considering the temporal context, we propose to minimize the following temporal ranking loss:

$$\mathcal{L}_{\text{tem}}^{k,j} = max\left(0, max\left(\Gamma(V', Q_k), \Gamma(V, Q'_j)\right) - \Gamma(V, Q_k) + \Delta_t\right), \tag{5.6}$$

where $\Gamma(V, Q_k) = \frac{1}{T}\sum_{t=1}^{T} w_t S(V_t, Q_k)$ indicates the query specific bag score computed by weighted sum of query specific frame scores. $\mathcal{L}_{\text{tem}}^{k,j}$ encourages video V and its paired query Q_k to be better aligned than a query Q'_j in negative query set Q'. Δ_t serves as similarity margin for temporal-level grounding. Finally, the temporal video-query ranking loss is defined as the average over the entire unaligned query set:

$$\mathcal{L}_{\text{tem}}^{k} = \frac{1}{K'}\sum_{j=1}^{K'} \mathcal{I}(Q_k \neq Q'_j)\mathcal{L}_{\text{tem}}^{k,j}. \tag{5.7}$$

The temporal MIL allows the queries to find the frames where they occur in a given video.

5.3.3.3 Overall objective function

In an untrimmed raw video, query does not necessarily occur in every frame of a video. Previous work Shi et al. (2019) propose a contextual similarity to weight the importance of frames corresponding to a query. In our work, we have the temporal MIL to learn the query specific attention over the temporal domain. Thus, only the query-related frame should contribute to the spatial grounding. We propose to utilize our temporal level grounding results as guidance to mask out query-irrelevant frames' contributions in spatio-level MIL ranking loss:

$$\mathcal{L}^k_{\text{spatio}} = \sum_{t=1}^{T} \mathcal{I}\ (S(V_t, Q_k) > 0)\, \mathcal{L}^{t,k}_{rank}, \tag{5.8}$$

where $\mathcal{L}^{t,k}_{rank}$ is computed by Eq. 5.4. The temporal grounding result $\mathcal{I}\ (S(V_t, Q_k) > 0)$ is incorporated into the spatial ranking loss so that spatial and temporal MILs are mutually guided.

We add a penalty term to avoid the trivial solution $S(V_t, Q_k) = 0$. The final objective function of our model is formulated as:

$$\mathcal{L}^k = \mathcal{L}^k_{\text{spatio}} + \mathcal{L}^k_{\text{tem}} + \frac{\lambda}{T} \sum_{t=1}^{T} -w_t S(V_t, Q_k), \tag{5.9}$$

where λ is the weight for the sparsity constraint. And the final objective is the average of ranking loss on each query and is summarized as $\mathcal{L} = \frac{1}{K} \sum_{k \in K} \mathcal{L}^k$.

5.4 Experiments

Following Shi et al. (2019), we train and evaluate our model on YouCookII dataset Zhou et al. (2018b) in a weakly-supervised setting. Besides, we validate the generalization ability of our model on RoboWatch dataset Sener et al. (2015).

5.4.1 Dataset

YouCookII Zhou et al. (2018b) contains $2,000$ cooking videos from 89 recipes. Each video recipe consists of 3 to 15 steps. Each step is described by a sentence including multiple queries. We follow Zhou et al. (2018a); Shi et al. (2019) to extract 15K video-description pairs from the steps. Training, validation and testing splits contain 5161, 3483 and 1560 pairs, respectively. The average duration of each step is 19.6s. Bounding box annotations Zhou et al. (2018a) for the most 67 frequently appearing objects in the description sentence for the validation and testing split are used. The presence and bounding boxes of objects are labeled every second in a video, which can be used to evaluate spatio-temporal grounding models.

RoboWatch Sener et al. (2015) contains 255 YouTube instructional videos, each of which also contains multiple steps. Huang et al. (2018) extends the bounding box annotation for a part of those videos, and the query can be either a word or a phrase. We follow Shi et al. (2019) to evaluate the generalization ability of our model trained on YouCookII Zhou et al. (2018a) dataset. Following Shi et al. (2019), we evaluate our model on the aligned pairs of video and query in RoboWatch. Since each query appears in all of the annotated frames of its video, we only evaluate spatial grounding on RoboWatch dataset.

5.4.2 Evaluation Metric

We follow Shi et al. (2019); Zhou et al. (2018a); Chen et al. (2019a) to evaluate spatial grounding performance using *box accuracy* and *query accuracy*. The *box accuracy* is defined as the ratio of correctly grounded boxes to all of the grounded boxes by setting a threshold, *i.e.,* , 50%, for Intersection-over-Union (IoU) between the grounded box and its corresponding ground-truth. *Query accuracy* is defined as the ratio of correctly grounded queries to all queries. Following Shi et al. (2019), the average of each class accuracy and the global accuracy without considering the class are evaluated, which are denoted as *macro-accuracy* and *micro-accuracy*, respectively. In addition, we follow an existing temporal grounding method Mithun et al. (2019) and compute the temporal IoU (tIOU) between the grounded and ground-truth temporal occurrence as the temporal grounding metric.

However, in previous work Shi et al. (2019); Zhou et al. (2018a), the *box accuracy* and *query accuracy* consider only the frames with query occurring. These two evaluation metrics ignore the frames that no query object appears, and thus are not suitable for evaluating the performance of spatio-temporal grounding on untrimmed videos. We propose the following metric to evaluate spatio-temporal grounding models for untrimmed videos:

$$\text{stACC} = \frac{1}{\left|\mathcal{S}^{(U)}\right|} \sum_{t \in \mathcal{S}^{(I)}} \mathcal{I}\left(\text{IoU}(\hat{r}^t, r^t) > R\right), \tag{5.10}$$

where $\mathcal{S}^{(U)}$ is the union set of frames in which either ground-truth or the grounded bounding boxes are located for a query in an entire video. $\mathcal{S}^{(I)}$ is the intersection set of frames in which both the ground-truth and the grounded bounding boxes occur simultaneously for a query. To compute the intersection, for each grounded box, we count the intersected box by computing the IoU between the grounded box $\hat{r}^t$ and its corresponding ground truth r^t with threshold R. Similar to existing grounding metrics, our proposed stACC can be used to compute *box accuracy* and *query accuracy* by considering the class of each query. Function $\mathcal{I}(\cdot)$ is an indicator function. The proposed metric in Eq. 5.10 will be used for evaluating the spatio-temporal grounding performance.

5.4.3 Implementation Details

Following Shi et al. (2019), the description sentence is parsed by Stanford CoreNLP parser Manning et al. (2014) into nouns and pronouns. We also parse the predicates of nouns/pronouns in the description sentence by SpaCy Honnibal and Johnson (2015). A pre-trained BERT Devlin et al. (2019) model is applied to encode the sentence. For visual modality, a Faster R-CNN framework Ren et al. (2015) with VGG-Net Simonyan and Zisserman (2014) as backbone pre-trained on Visual Genome Krishna et al. (2017) is applied to extract top-20 confident region proposals for each frame, which is the same setting as Shi et al. (2019); Zhou et al. (2018a). We uniformly sample 16 frames from each video. Hyperparameter v_a in spatial prior is set to 40. Visual and textual features are embedded to a joint feature space with 512-dimension. $tanh$ is used as the activation function for both visual and text embedding.

We use TITAN Xp and implement the network using PyTorch. Adam with learning rate 0.001 is used for optimization. The ranking margin Δ_t and Δ_s is set to 10 and 5. Constraint weight λ is set to 0.9. We use a batch-size of 8 in all experiments. Thus, each of positive sample is coupled with 7 negative samples. Following Shi et al. (2019); Zhou et al. (2018a); Chen et al. (2019a), we report the grounding results in both validation and test split.

5.4.4 Comparison

5.4.4.1 Spatial Grounding on YouCookII Dataset

We compare our method with the state-of-the-art weakly-supervised video grounding methods Chen et al. (2019b); Shi et al. (2019); Zhou et al. (2018a) and two extensions from image grounding methods *DVSA* Karpathy and Fei-Fei (2015) and *GroundR* Rohrbach et al. (2016). Following Shi et al. (2019); Zhou et al. (2018a), we utilize RPN to extract region proposals. Existing work Shi et al. (2019); Zhou et al. (2018a) only evaluate the spatial grounding accuracy on the frames where the query occurs. The frames without the query are disregarded in evaluation. We follow this evaluation setting and report the results under the four metrics used in Shi et al. (2019). As shown in Table 5.1, the proposed method on spatial grounding consistently outperforms the comparison methods. This is because we bridge the granularity gap between text and visual domains by considering the activity-effect. In addition, our spatio-temporal MIL framework ensures a spatial grounding model learned from the frames that query appears, even without temporal annotations.

		box accuracy(%)				query accuracy(%)			
Methods	pre-trained	macro		micro		macro		micro	
		val	test	val	test	val	test	val	test
Extended GroundR	MSCOCO	19.63	19.94	-	-	-	-	-	-
Zhou et al.	MSCOCO	30.31	31.73	-	-	-	-	-	-
Chen et al.	MSCOCO	33.24	34.90	-	-	-	-	-	-
Extended DVSA	VisualGenome	36.90	37.55	44.26	44.16	38.48	39.31	46.27	46.41
Shi et al.	VisualGenome	39.54	40.71	46.41	46.33	41.29	42.45	48.52	48.41
Ours (BERT)	VisualGenome	37.40	38.88	48.12	45.20	39.00	40.55	46.10	47.23
Ours (Glove+Activity)	VisualGenome	38.50	39.28	46.57	45.85	40.11	41.07	48.59	47.91
Our full model	VisualGenome	**40.66**	**41.67**	**49.11**	**48.22**	**41.43**	**42.55**	**49.71**	**48.91**

Table 5.1 Weakly-supervised spatial grounding results on YouCookII. “pre-trained” indicates the dataset that RPN is pre-trained on. Zhou et al., extended GroundR and Chen et al. only report macro box accuracy in their papers.

5.4.4.2 Temporal Grounding on YouCookII Dataset

We compare our method with an existing weakly-supervised video temporal grounding method TGA Rohrbach et al. (2016) and an extension of Shi et al. (2019) to temporal grounding. The extension of Shi et al. (2019) to temporal grounding is achieved by extending its frame-query

Methods	tIOU
TGA Mithun et al. (2019)	29.43
Extension of Shi et al. (2019)	27.12
Ours	**39.51**

Table 5.2 Weakly-supervised temporal grounding results on YouCookII. tIOU is used as the evaluation metric.

contextual similarity module, which conducts 0-1 normalization of frame-query importance across frames during training. We use it in the test phase to mask out the frame-query pair whose contextual similarity score is less than 0.5.

As shown in Table 5.2, our approach significantly outperforms the extension of Shi et al. (2019) and TGA Mithun et al. (2019) by 12% and 10%, respectively. This is because these video temporal grounding method grounds a query to the relevant frames based on the similarity between the query and the entire frame. In our method, spatial grounding provides guidance for temporal grounding to be focused on the query specific region. This allows us to represent the visual data more accurately.

5.4.4.3 Spatio-temporal Grounding on YouCookII Dataset

Since there is no existing weakly-supervised spatio-temporal grounding method for untrimmed videos, we extend Shi et al. (2019) to ground temporal occurrences (Extension of Shi et al. (2019)) using the method described above. We also compare with the weakly-supervised spatio-temporal grounding method Chen et al. (2019b), which is originally developed for trimmed video. The performance of spatio-temporal grounding methods is evaluated using the metric stACC described in Eq. 5.10.

As shown in Table 5.3, our approach significantly outperforms the Extension of Shi et al. (2019) by $5 \sim 8\%$. This shows that a direct extension to spatio-temporal grounding is far from solving this challenging problem. Our method is more effective since we solve this problem using a mutually guided MIL. When generalized to untrimmed videos, Chen et al. (2019b) shows inferior results to ours, because they highly rely on a visual tracker that easily fails due to the frequent camera shot cut in untrimmed videos.

Methods	macro		micro	
	val	test	val	test
Extension of Shi et al. (2019)	15.89	19.10	17.35	18.54
Chen et al. (2019b)	7.31	7.70	8.02	8.79
Ours-max	5.17	5.25	5.93	6.11
Ours w/o attention	18.94	20.98	22.31	21.41
Our full model	**21.73**	**24.25**	**25.50**	**25.65**

Table 5.3 Weakly-supervised spatio-temporal grounding results on YouCookII. stACC is used as the evaluation metric.

Methods	all	unseen split
Extended DVSA Karpathy and Fei-Fei (2015)	28.25	25.12*
Shi et al. (2019)	31.68	26.79*
Ours w/o activity	30.11	26.56
Our full model	**34.21**	**35.97**

Table 5.4 Generalization results on RoboWatch using query micro-accuracy (%). "*" indicates the results are achieved by running the authors' code on our side. All the other comparison results are from their original papers.

5.4.4.4 Generalize Grounding Model to RoboWatch Dataset

Following Shi et al. (2019), we conduct the generalization ability experiment of the grounding model trained on YouCookII dataset. We train our grounding model using the nouns and pronouns parsed in the sentences of YouCookII and directly test the grounding model on RoboWatch dataset. We compare our method with two existing methods Shi et al. (2019) and extended DVSA Karpathy and Fei-Fei (2015) and a variant of our method that does not contain activity-driven object states encoding module The comparison is conducted on two types of data split, including the entire test set of RoboWatch and its unseen split which only consists of the objects that never occur in YouCookII such as "oreo", "flesh", "alcohol", "hanger", "tie" *etc*.

As shown in Table 5.4, the proposed activity-driven model outperforms the variant Ours w/o activity and two other existing methods Shi et al. (2019) and the extended DVSA Karpathy and Fei-Fei (2015) by a large margin. Also, our full model's performance in the unseen split is even better than the performance in the entire test set denoted as "all". This is because we model the activity effect on objects' physical states. Thus, even though our model has never seen the query during training, it can utilize the seen activity information to ground the unseen query on which the activity is performed.

5.4.5 Ablation Studies

5.4.5.1 Activity-driven Object-States Encoding

We conduct ablation study on the activity-driven object states encoding module with following two variants: 1) "**BERT**" which first encodes the entire description sentence by a pre-trained BERT model Devlin et al. (2019) and then extracts the query embedding of the objects without considering activity; 2) "**Glove+Activity**". It first extracts the predicates and nouns/pronouns from the description sentence by Manning et al. (2014) and then encodes the predicate-object pair into 200-dimensional GloVe Pennington et al. (2014). Note that GloVe is used as word embedding in Shi et al. (2019); Zhou et al. (2018a).

As shown in Table 5.1, the superiority of our full model over the variant "BERT" shows that encoding activity effect on object states benefits grounding. As expected, the activity-driven object state encoding module proves to bridge the granularity gap between the coarse text modality and the rich visual modality, by incorporating the underlying activity-effect on object states into text representation. Moreover, the performance gain from the activity cue is significantly larger than the gain from better query embeddings, *i.e.,* , Glove and BERT features. This further demonstrates the effectiveness of the proposed activity-driven encoding.

5.4.5.2 Temporal MIL Loss

We conduct ablation study on the temporal grounding with the following variants: 1) "**Ours-max**", which replaces the average instance operation in Eq. 5.2 by max operation, selecting the top-ranked query specific frame to represent the bag; 2) "**Ours w/o attention**", which uses Eq. 5.5 as the temporal ranking loss but removes the guidance of frame consistency attention block.

Table 5.3 shows that our full model achieves the best performance. Its superiority over "Ours w/o attention" demonstrates that the temporal context information w.r.t frame similarity plays an important role in video grounding with large visual inconsistency. The variant "Ours-max" is inferior to others, indicating selecting the top-ranked frame as the video representation is not appropriate for temporal grounding. This is because in an untrimmed video, the query may appear discontinuously across frames, leading to multiple frames for the query in the video.

Methods	distant view split		all	
	box	query	box	query
Ours w/o region refine	20.54	21.35	48.07	48.72
Our full model	**21.03**	**21.85**	**48.22**	**48.91**

Table 5.5 Ablation study on YouCookII for our activity-driven region proposal refinement module. The results are box/query micro-accuracy. Results of the distant view frame split in the test set and the entire test set are reported.

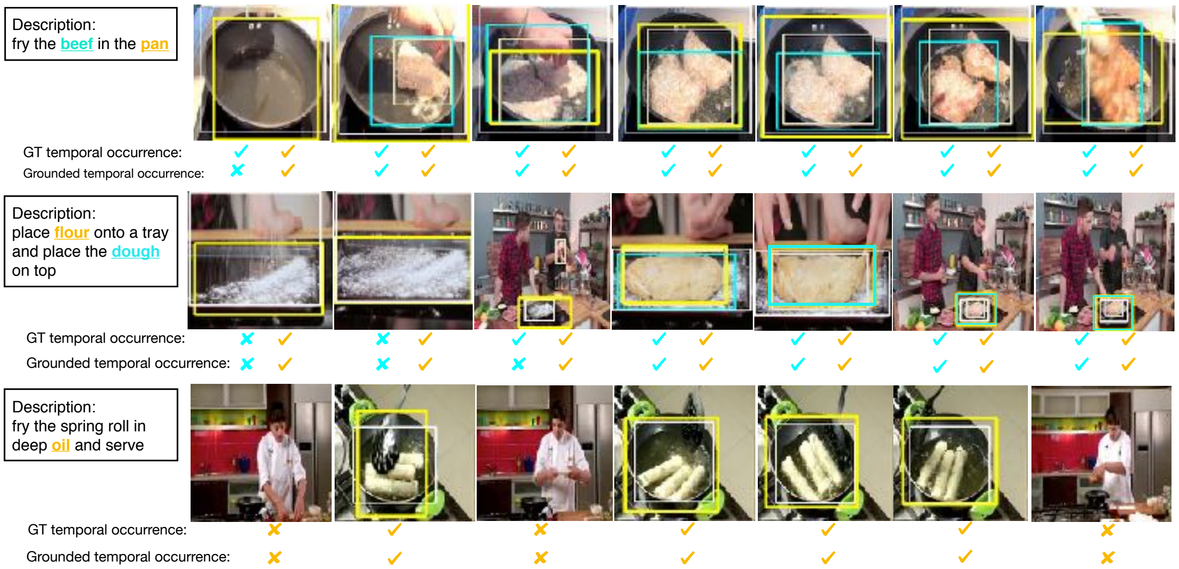

Description:
fry the beef in the pan
GT temporal occurrence:
Grounded temporal occurrence:
Description:
place flour onto a tray and place the dough on top
GT temporal occurrence:
Grounded temporal occurrence:
Description:
fry the spring roll in deep oil and serve
GT temporal occurrence:
Grounded temporal occurrence:

Figure 5.4 Qualitative results of our spatio-temporal video grounding model. The yellow and cyan boxes are the grounded results of the corresponding queries in description sentences. The white boxes are their ground-truth. Best viewed in color.

5.4.5.3 Region Proposal Refinement

We conduct ablation study for the activity-driven region proposal refinement module. On YouCookII dataset, only 22% frames are captured distant view. Thus, we evaluate this module on the distant view split that only contains the frames with human and the entire test set, correspondingly. In the variant "Ours w/o region refine", we simply keep the top-N proposals without refinement. As shown in Table 5.5, our method outperforms the variant without region refinement, which elaborates the effectiveness of our region refinement module. Our full model refines the proposals to include more query related proposals. This makes the query more likely to be grounded.

5.4.6 Qualitative Results

The qualitative results of YouCookII dataset are shown in Fig. 5.4. Each row depicts 6 frames sampled from a video. Camera shot cut frequently appears in these videos. But even though the large visual inconsistency appears, our method is able to ground each query in terms of its temporal occurrence and spatial locations.

5.5 Summary

In this paper, we investigate the spatio-temporal grounding in untrimmed videos with frequent visual inconsistency in a weakly-supervised manner. We develop two novel MIL ranking losses for the spatial and temporal domains. Furthermore, to bridge the granularity gap between the coarse text information and the detailed visual information, we introduce an activity-driven object state encoding module to enhance textual representation. Experiments on two popular datasets demonstrate the superiority of our method and its generalization ability to other datasets with unseen queries.

BIBLIOGRAPHY

Alomari, M., Duckworth, P., Hogg, D. C., and Cohn, A. G. (2017). Natural language acquisition and grounding for embodied robotic systems. In *Thirty-First AAAI Conference on Artificial Intelligence*.

Cadene, R., Ben-Younes, H., Cord, M., and Thome, N. (2019). Murel: Multimodal relational reasoning for visual question answering. In *CVPR*.

Cao, Z., Hidalgo, G. M., Simon, T., Wei, S., and Sheikh, Y. (2019). Openpose: Realtime multi-person 2d pose estimation using part affinity fields. *TPAMI*.

Chen, K., Gao, J., and Nevatia, R. (2018). Knowledge aided consistency for weakly supervised phrase grounding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4042–4050.

Chen, L., Zhai, M., He, J., and Mori, G. (2019a). Object grounding via iterative context reasoning. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 0–0.

Chen, Z., Ma, L., Luo, W., Tang, P., and Wong, K.-Y. K. (2020). Look closer to ground better: Weakly-supervised temporal grounding of sentence in video. *arXiv preprint arXiv:2001.09308*.

Chen, Z., Ma, L., Luo, W., and Wong, K.-Y. K. (2019b). Weakly-supervised spatio-temporally grounding natural sentence in video.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). Bert: Pre-training of deep bidirectional transformers for language understanding. In *ACL*, pages 4171–4186.

Duan, X., Huang, W., Gan, C., Wang, J., Zhu, W., and Huang, J. (2018). Weakly supervised dense event captioning in videos. In *Advances in Neural Information Processing Systems*, pages 3059–3069.

Fan, H., Lin, L., Yang, F., Chu, P., Deng, G., Yu, S., Bai, H., Xu, Y., Liao, C., and Ling, H. (2019). Lasot: A high-quality benchmark for large-scale single object tracking. In *CVPR*.

Gao, M., Davis, L., Socher, R., and Xiong, C. (2019). Wslln: Weakly supervised natural language localization networks. In *EMNLP-IJCNLP*.

He, K., Gkioxari, G., Dollár, P., and Girshick, R. (2017). Mask R-CNN. In *ICCV*.

Honnibal, M. and Johnson, M. (2015). An improved non-monotonic transition system for dependency parsing. In *EMNLP*.

Huang, D.-A., Buch, S., Dery, L., Garg, A., Fei-Fei, L., and Niebles, J. C. (2018). Finding "it": Weakly-supervised, reference-aware visual grounding in instructional videos. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Karpathy, A. and Fei-Fei, L. (2015). Deep visual-semantic alignments for generating image descriptions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3128–3137.

Krishna, R., Zhu, Y., Groth, O., Johnson, J., Hata, K., Kravitz, J., Chen, S., Kalantidis, Y., Li, L.-J., Shamma, D. A., et al. (2017). Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision*, 123(1):32–73.

Kwak, S., Cho, M., Laptev, I., Ponce, J., and Schmid, C. (2015). Unsupervised object discovery and tracking in video collections. In *ICCV*.

Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J. R., Bethard, S., and McClosky, D. (2014). The stanford corenlp natural language processing toolkit. In *ACL*.

Mithun, N. C., Paul, S., and Roy-Chowdhury, A. K. (2019). Weakly supervised video moment retrieval from text queries. In *CVPR*.

Pennington, J., Socher, R., and Manning, C. D. (2014). Glove: Global vectors for word representation. In *EMNLP*.

Prest, A., Leistner, C., Civera, J., Schmid, C., and Ferrari, V. (2012). Learning object class detectors from weakly annotated video. In *CVPR*.

Ren, S., He, K., Girshick, R., and Sun, J. (2015). Faster r-cnn: Towards real-time object detection with region proposal networks. In *NIPS*.

Rohrbach, A., Rohrbach, M., Hu, R., Darrell, T., and Schiele, B. (2016). Grounding of textual phrases in images by reconstruction. In *European Conference on Computer Vision*, pages 817–834. Springer.

Sener, O., Zamir, A. R., Savarese, S., and Saxena, A. (2015). Unsupervised semantic parsing of video collections. In *ICCV*.

Shi, J., Xu, J., Gong, B., and Xu, C. (2019). Not all frames are equal: Weakly-supervised video grounding with contextual similarity and visual clustering losses. In *CVPR*, pages 10444–10452.

Shridhar, M. and Hsu, D. (2018). Interactive visual grounding of referring expressions for human-robot interaction. *arXiv preprint arXiv:1806.03831*.

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

Wang, N., Song, Y., Ma, C., Zhou, W., Liu, W., and Li, H. (2019a). Unsupervised deep tracking. In *CVPR*.

Wang, X., Jabri, A., and Efros, A. A. (2019b). Learning correspondence from the cycle-consistency of time. In *CVPR*.

Yang, S., Li, G., and Yu, Y. (2019a). Cross-modal relationship inference for grounding referring expressions. In *CVPR*.

Yang, Z., Gong, B., Wang, L., Huang, W., Yu, D., and Luo, J. (2019b). A fast and accurate one-stage approach to visual grounding. In *ICCV*.

Yang, Z., Kumar, T., Chen, T., and Luo, J. (2019c). Grounding-tracking-integration. *arXiv preprint arXiv:1912.06316*.

Yu, H. and Siskind, J. M. (2013). Grounded language learning from video described with sentences. In *ACL*.

Zhou, L., Kalantidis, Y., Chen, X., Corso, J. J., and Rohrbach, M. (2019). Grounded video description. In *CVPR*.

Zhou, L., Louis, N., and Corso, J. J. (2018a). Weakly-supervised video object grounding from text by loss weighting and object interaction.

Zhou, L., Xu, C., and Corso, J. J. (2018b). Towards automatic learning of procedures from web instructional videos. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

CHAPTER 6

EXPLAINABLE VIDEO ENTAILMENT WITH VISUALLY GROUNDED EVIDENCE

6.1 Introduction

Bridging the gap between computer vision and natural language processing is a rapid growing research area in various tasks including visual captioning Zhou et al. (2020); Vinyals et al. (2015), VQA Lei et al. (2018); Antol et al. (2015); Tapaswi et al. (2016), and visual-textual retrieval Lei et al. (2020); Li et al. (2019). Liu et al. (2020) introduced a new video entailment problem to infer the semantic entailment between a premise video and a textual hypothesis. As shown in Fig. 6.1, video entailment Liu et al. (2020) task aims at determining whether a textual statement is *entailed* or *contradicted* by a video. In Fig. 6.1, the label for the first statement with the premise is *entailment* because the statement can be concluded from the dialog of the first clip in which "the woman wearing jeans" appears. On the contrary, the second statement is labeled as *contradiction*, because the premise does not have evidence to conclude the statement. In this paper, we aim to address the video entailment with a faithful explanation.

The main challenge of video entailment is that it requires fine-grained reasoning to understand the complex story-based videos and then make a correct judgment. The story-based videos are also accompanied by the textual dialog (subtitles) (see Fig. 6.1). In the existing method for video entailment Liu et al. (2020), video frames are less exploited than dialog, because it lacks of a fine-grained understanding of the video and the model does not know which frames in the long video are related to the statement. However, the entities in the textual statement are usually people with their attributes, *e.g.,* , "A woman wearing jeans" (see Fig. 6.1), which should be implied in the video frames instead of the dialog.

To this end, we propose to enhance the entailment judgment by introducing a visual grounding model that links the entity described in a statement to the evidence in the video. This is motivated by the fact that the statement is usually only related to a small subset of the long and untrimmed video. Based on this, a visual grounding module for the entities described in the statement is developed to localize the clips where the entity appears and guide the judgment to focus on the entity's occurring

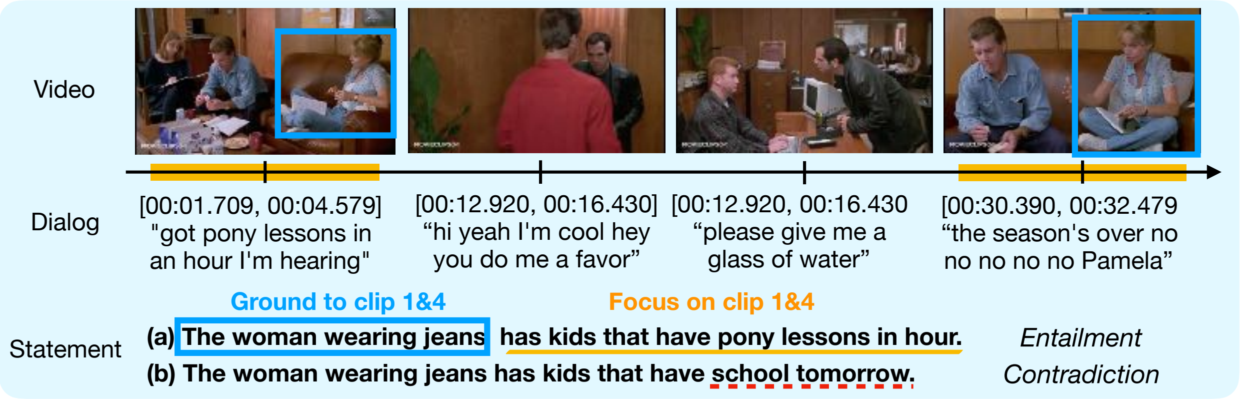

Video
Dialog
Statement
[00:01.709, 00:04.579] "got pony lessons in an hour I'm hearing"
[00:12.920, 00:16.430] "hi yeah I'm cool hey you do me a favor"
[00:12.920, 00:16.430 "please give me a glass of water"
[00:30.390, 00:32.479 "the season's over no no no no Pamela"
Ground to clip 1&4
Focus on clip 1&4
(a) The woman wearing jeans has kids that have pony lessons in hour.
Entailment
(b) The woman wearing jeans has kids that have school tomorrow.
Contradiction

Figure 6.1 Video entailment aims at judging if a statement is entailed or contradicted by a video and its aligned textual dialog. A pair of real and fake statements have a similar structure and subtle difference (marked by the red dot line). We incorporate visual grounding into the entailment judgment. The entity grounding, *e.g.,* , "A woman wearing jeans" guides the entailment judgment module to focus on the entity-relevant frames and the corresponding sentences in the dialog (marked by blue in the temporal axis) to make a correct judgment. Best viewed in color.

clips as well as the aligned sentences in the dialog. For example, the statements in Fig. 6.1 are linked to the first and fourth clips and sentences, considering the entity "The woman wearing jeans". By highlighting the relevant clips and sentences, the details can be better understood compared to Liu et al. (2020) that does not have grounding guidance and equally considers all of the frames.

Visual grounding has been attempted in many video+language tasks, such as image captioning Zhou et al. (2019) and VQA Lei et al. (2019). However, it cannot be directly generalized to the entailment task, because the bounding box annotations of grounding are not provided in the entailment dataset. Therefore, we resort to the existing weakly-supervised object grounding methods Karpathy and Fei-Fei (2015); Chen et al. (2018) to address the training of the grounding module. But these methods are limited to explicit natural objects (*e.g.,* , "apple", "river"). Our grounding is more demanding, as we target at the described entities with fine-grained attributes, such as hair, clothes and gender, to be grounded to the challenging story-telling videos.

Furthermore, we aim at improving the faithfulness of the entailment model by evaluating if the entailment is judged based on correct evidence. A faithful entailment model should tell not only *whether the statement is contradictory to the video* but also *which words or phrases in the statement make it contradictory to the video*. A pair of real/fake statements usually have a similar structure and only have very subtle differences, with only a small number of words' replacement, *e.g.,* . "pony lessons in hour" and "school tomorrow" marked by the red dot line in Fig. 6.1. Thus, we propose to regularize the training of the entailment judgment module by encouraging the local explanation on the contribution of the words in the statement to conform to the subtle difference.

Our main contribution is threefold. First, we propose a novel approach to address video entailment with visually grounded evidence. Second, we exploit the pairwise real/fake statements to add the explainability to the entailment model, which can tell the specific words or phrases that make the statement contradictory to the video. Third, extensive results demonstrate that our method outperforms the state-of-the-art video entailment method.

6.2 Related Work

6.2.1 Visual Entailment

Natural language inference Dagan et al. (2005); Condoravdi et al. (2003); MacCartney and Manning (2009); Camburu et al. (2018) is the task of understanding if a hypothesis sentence is entailed or contradicted by a premise sentence, which is a fundamental task in natural language understanding. Inspired by the textual entailment, recently visual entailment is proposed to extend NLI to the visual domain. In visual entailment, the premise is an image or a video. And the goal is to predict if the textual hypothesis can be confirmed in the visual premise.

Recently, researchers began to solve visual entailment mainly on image premise. SNLI-VE Xie et al. (2019) is a visual entailment dataset combining the textual entailment Bowman et al. (2015) and Flickr30k image caption Young et al. (2014). It also provides a solution model that utilizes ROI generation and models the fine-grained cross-modal information. However, the hypothesis (*e.g.,* , "The two women are holding packages") is much more straightforward compared to the hypothesis in our video entailment. e-SNLI-VE-2.0 Do et al. (2020) appends and corrects SNLI-VE Xie et al. (2019) by the human-written language hypothesis. It also provides the explanation ground-truth of why the hypothesis is entailed/contradicted by the premise. NLVR2 Suhr et al. (2019) is another image entailment dataset that requires quantitative and comparing reasoning. But similar to SNLI-VE Xie et al. (2019), it also mainly focuses on objects in the natural images.

Recently, Liu et al. (2020) proposed Violin dataset that focuses on video entailment. Video entailment is a challenging task as the complex temporal dynamics occur in the video. A fine-grained reasoning of the social relations, human motions and intentions is necessary to understand the story-based content and make a correct judgment.

6.2.2 Grounding for Video+Language Reasoning

Recently, many video+language tasks have been trying to explicitly link the language sentence to the evidence in the video. Zhou et al. (2019) proposed a video description dataset with the annotation of the bounding boxes of the referred objects. With this dataset, a good captioning model is desirable by attending to appropriate video regions. For video question answering, Lei et al. (2019) built a dataset with the spatio-temporal grounding annotation, which requires the model to localize the temporal moments, detected the referred object, and answer the questions.

Different from captioning and VQA, video entailment needs a fine-grained understanding of the entities with detailed attributes. Meanwhile, the existing video entailment does not provide the grounding annotation. Thus, we propose to achieve the entity grounding in a weakly-supervised manner.

6.2.3 Weakly-supervised Entity Grounding

Visual grounding is to localize the described entity to its occurring regions visually. Since the annotation of bounding boxes is very expensive, sundry efforts have been made to achieve object grounding in a weakly-supervised manner Karpathy and Fei-Fei (2015); Chen et al. (2018); Rohrbach et al. (2016), mainly based on multiple instance learning. It also has been extended to video domain Zhou et al. (2018); Shi et al. (2019); Huang et al. (2018); Chen et al. (2019b,a, 2020), to achieve spatio-temporal grounding of entities in an untrimmed video.

In the video entailment task, the visually related entities are mainly characters, while the existing grounding methods aim at grounding natural objects. Our grounding requires a fine-grained understanding of human gender, dress, hair and other attributes. Therefore, we cannot directly generalize the existing grounding methods to video entailment.

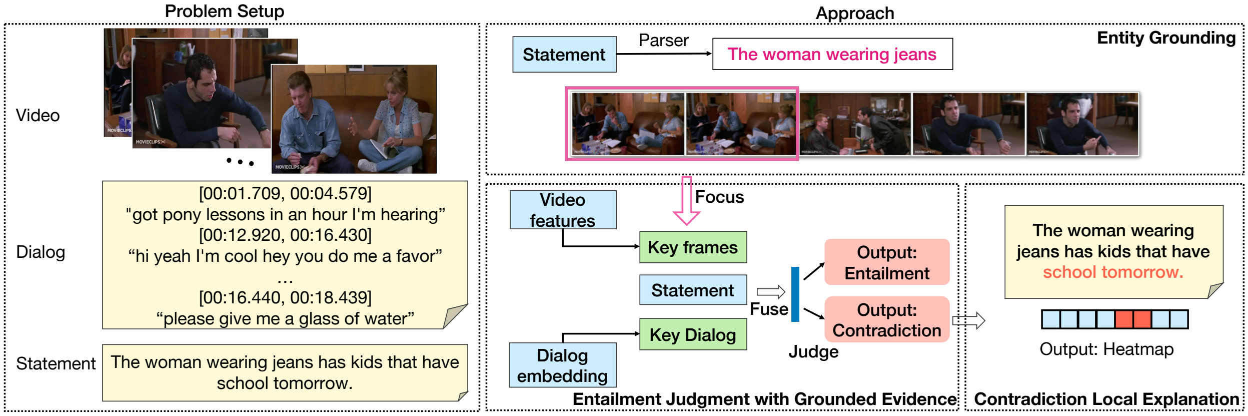

Problem Setup
Video
Dialog
[00:01.709, 00:04.579]
"got pony lessons in an hour I'm hearing"
[00:12.920, 00:16.430]
"hi yeah I'm cool hey you do me a favor"
...
[00:16.440, 00:18.439]
"please give me a glass of water"
Statement
The woman wearing jeans has kids that have school tomorrow.
Approach
Statement
Parser
The woman wearing jeans
Entity Grounding
Video features
Focus
Key frames
Statement
Key Dialog
Dialog embedding
Fuse
Output: Entailment
Output: Contradiction
Judge
Entailment Judgment with Grounded Evidence
The woman wearing jeans has kids that have school tomorrow.
Output: Heatmap
Contradiction Local Explanation

Figure 6.2 Given a video, its aligned dialog in text, and a textual statement for the video as input, our goal is to predict if the statement is *entailed* or *contradicted* by the video and dialog. Our model consists of three sub-networks: Entity Grounding, Entailment Judgment with Grounded Evidence, and Contradiction Local Explanation. The entity grounding module helps to find if the described entity occurs in the video clips. Moreover, entity grounding guides the judgment module to focus on the entity-relevant clips and the corresponding sentences in the dialog (marked as "Key"), to make a correct judgment. If judged as "contradiction", our model can also explain which words or phrases in the statement make it contradictory to the video by generating an explanation heatmap.

6.2.4 Multi-modal VQA

Different from image entailment, video entailment is supposed to understand story-based video content, such as movies. This is more challenging than the plain videos as multiple factors such as human interactions, emotions, motivation, and scenes appear. Similar to existing videoQA datasets Lei et al. (2019, 2020), the input to our entailment task is multi-modal, including both videos and textual subtitles. For multi-modal VQA, early fusion was commonly used in merging different modalities Na et al. (2017). Recent methods mainly leverage late fusion approaches Kim et al. (2018, 2019). Another aspect Kim et al. (2020) is to utilize the content of QA pairs to shift to the relevant modality and constrain the contribution of the irrelevant ones.

Video entailment requires a fine-grained understanding. The statement may only relate to the details in a long and untrimmed video. Thus, we propose to ground the described entities to their occurring clips and highlight the dialog sentence aligned to those clips for entailment judgment.

6.3 Our Approach

Given a story-like video aligned with a textual dialog (subtitles) and a hypothesis statement, the entailment task is to predict if the hypothesis statement is *entailed* or *contradicted* by the premise video (see the left of Fig. 6.2). The right part of Fig. 6.2 shows the overall pipeline of the proposed method. We decompose our model into three sub-networks: entity grounding, entailment judgment with grounded evidence, and contradiction local explanation, to address entailment in a modularized manner.

The motivation of grounding entities described in the statement (*e.g.,* , "a woman wearing a red cape") to frames comes from the observation that video modality is not well exploited compared to dialog modality in the existing method Liu et al. (2020). However, many contradictory statements such as the incorrect attributes should be determined from the frames instead of the dialog, (*e.g.,* , "a woman wearing a blue cape") in Fig. 6.3. Moreover, the statements are written about different aspects of a video Liu et al. (2020), and a statement is usually related to a small subset of video frames. The entity grounding helps to find the entity-relevant frames and then guides the entailment judgment module to highlight these frames. To learn a credible entailment judgment model, we propose to not only judge the semantic entailment but also explain which words or phrases make the statement contradictory to the video by a heatmap that indicates the contribution of each word in the statement to the model prediction.

6.3.1 Preliminaries

Text Representation. Following Violin Liu et al. (2020), we use BERT encoder Devlin et al. (2019a) provided by Violin to represent the statement and dialog, resulting in a 768-dimension vector for each word. Then using a bi-directional LSTM for both statement and dialog, each word is also embedded to d-dimension. A statement is tokenized into a word sequence, in the length of N_l. A textual dialog is also tokenized and represented as a word sequence. Then, by encoding, the statement is represented as $R = \{r_i\}_{i=1}^{N_l}$ in which r_i indicates the i-th word's representation. The dialog is represented as $H = \left\{h_j\right\}_{j=1}^{N_s}$, in which h_j indicates the j-th word's representation. N_s denotes the number of words in the long dialog. The starting time t_s^j and the ending time t_e^j of the j-th sentence are also provided, which can be aligned with the video frames.

Video Representation. Following Violin Liu et al. (2020), we extract a sequence of visual features from video frames and then encode the visual features by a bi-directional LSTM layer.

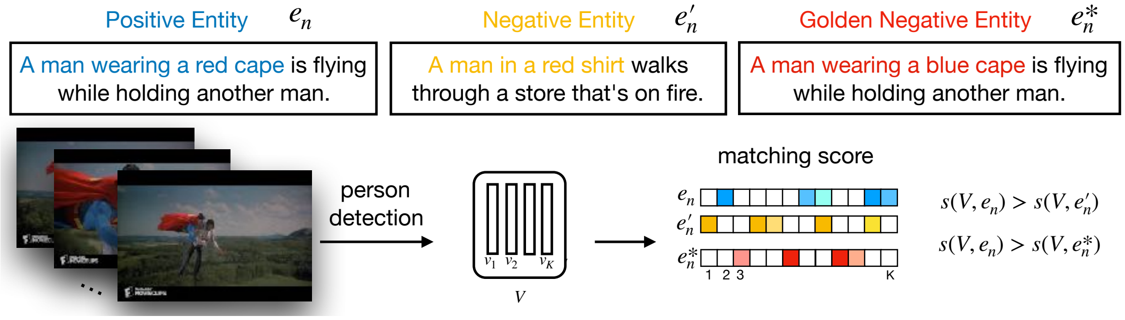

Positive Entity e_n
A man wearing a red cape is flying while holding another man.
Negative Entity e'_n
A man in a red shirt walks through a store that's on fire.
Golden Negative Entity e^*_n
A man wearing a blue cape is flying while holding another man.
person detection
v_1 v_2 v_K
V
matching score
e_n
e'_n
e^*_n
1 2 3
K
$s(V, e_n) > s(V, e'_n)$
$s(V, e_n) > s(V, e^*_n)$

Figure 6.3 Training of the entity grounding module. We extract the positive entity e_n from the statement aligned with the video and the negative entity e'_n from a statement unaligned with the video. Besides, real and fake statements are formed in pairs. Thus, the entity in the fake statement can be utilized as a golden negative entity e^*_n, which is slightly different from e_n and is a hard sample to enhance the grounding model's training. The training process encourages the matching score of the positive entity to be larger than any negative entity. Best viewed in color.

The video is then represented as $\mathbf{C} \in \mathbb{R}^{T \times d}$, where T is the number of frames, and d is the feature dimension of each frame.

To realize grounding, we first detect the people in the input video. Specifically, we extract the frames of the middle timestamps corresponding to each sentence $(t^j_s + t^j_e)/2$ and apply Faster R-CNN Ren et al. (2015) pretrained on COCO Lin et al. (2014) to detect all of the people from each frame and extract their features. Each person is represented by a 4096-dimension vector, denoted as v_k. Then each video is formed as a set of persons $V = \{v_k\}^K_{k=1}$, where v_k encodes the k-th person.

6.3.2 Entity Grounding Module

In the existing video entailment method Liu et al. (2020), the performance gain of video modality is limited compared to dialog modality. Visual information needs fine-grained understanding, but the existing work equally considers all of the frames even if the frames are not relevant to the statement. Video modality should be responsible for a lot of information described in the statement such as entity attributes (*e.g.,* , gender and clothes). We propose to leverage entity grounding in the video modality to improve the entailment judgment in a modularized manner (see Fig. 6.2). First, our grounding module is developed to achieve spatio-temporal grounding of the subject entity described in the statement. The predicted temporal occurrences of the entity are used to guide the following cross-modal entailment judgment.

However, two technical challenges need to be handled to leverage visual grounding for the entailment task. First, spatial-temporal annotations of entities are typically not available for the entailment task so that existing fully-supervised grounding-based VideoQA methods Lei et al. (2019) cannot be directly leveraged. We resort to multiple instance learning Zhou et al. (2018) to achieve entity grounding in a weakly-supervised fashion. Second, detailed visual attributes (*e.g.,* , clothes and hair) of entities are essential for the entailment task but they are typically ignored by the existing object grounding methods Shi et al. (2019); Zhou et al. (2018); Chen et al. (2020).

To extract the entity and its attributes from a textual statement, we employ a constitute parsing method Kitaev and Klein (2018). For example, in Fig. 6.2, “The woman wearing jeans” is an entity extracted from the corresponding statement “The woman wearing jeans has kids that have pony lessons in hour”. The extracted entities in a statement are denoted as $E = \{e_n\}_{n=1}^{N_e}$, where N_e is the total number of entities and e_n indicates the n-th entity.

To ground the entity to its occurring frames, we compute the matching score $s(V, e_n)$ between video V and an entity e_n as:

$$s(V, e_n) = \frac{1}{K} \sum_{k=1}^{K} \sigma(FC_1(v_k || e_n)) \tag{6.1}$$

where FC_1 is a fully-connected layer and σ is the sigmoid activation. We take average of the scores of the K people as the entity-video matching score $s(V, e_n)$.

Following the existing visual-textual matching work Li et al. (2019); Chen et al. (2020); Zhou et al. (2018), we formulate the weakly-supervised learning of grounding as:

$$\mathcal{L}_{ga} = -\log(1 - s(V, e_n')) - \log(s(V, e_n)), \tag{6.2}$$

where e_n' is a “negative entity” extracted from a randomly sampled statement from another video, which is different from e_n. Eq. 6.2 encourages that the aligned video-entity pair (V, e_n) to better matched and the unaligned pair $\left(V, e_n'\right)$ to be less matched.

Different from weakly-supervised video grounding Chen et al. (2020); Zhou et al. (2018), the entailment task consists of the real/fake statements in pairs. Thus, we have the opportunity to obtain hard negative samples, which is the entity described in the fake statement but NOT described in the real statement. As shown in Fig. 6.3, the negative version is “a man wearing a blue cape”, which is very similar to the positive one “a man wearing a red cape” but is contradicted by the video. We name it as “golden negative entities” e_n^* and use it in training the grounding module:

$$\mathcal{L}_{gb} = -\log(1 - s(V, e_n^*)) - \log(s(V, e_n)), \tag{6.3}$$

$\mathcal{L}_{gb}$ encourages the video V to match more to its aligned entity e_n and less to the golden negative entity e_n^*. To sum up, we train the grounding model by the grounding loss $\mathcal{L}_g$ which balances the negative entities and the golden negative entities by β.

$$\mathcal{L}_g = \mathcal{L}_{ga} + \beta \mathcal{L}_{gb}, \tag{6.4}$$

During the inference, if the matching score $s(v_k, e_n) = \sigma(FC_1(v_k || e_n))$ between a person v_k and an entity e_n exceeds a threshold, we consider that the k-th people is e_n. The temporal grounding result will be used to guide the entailment judgment in Sec 3.3.

6.3.3 Entailment Judgment with Grounded Evidence

Statements are usually related to a small subset of the video, instead of the entire video. For example, in Fig. 6.2, the clause in the statement “kids that have school tomorrow” should be judged from the first sentence in the dialog. Thus, we utilize the entity grounding result to highlight the frames and the corresponding textual dialog in the temporal range that the entity occurs, since the frames and dialog are aligned by temporal boundaries. The highlighted frame and dialog embeddings are concatenated and marked as key embedding C_O, H_O.

The model takes three streams in different modalities as input: video frames, dialog, and statements. We leveraged the visually grounded evidence to make our model fixate its attention on the frames where the entity appears. Then, we fuse the multi-modal data and predict whether the statement is entailed or contradicted by the video.

To bridge the modal discrepancy between the video frames and textual content, we use heterogeneous reasoning Zhang et al. (2019) to fuse the statement representation R with different context embedding, including video embeddings C, dialog embeddings H and key embeddings C_O, H_O (see Fig. 6.4) respectively. The heterogeneous reasoning is based on a graph convolution layer Chen

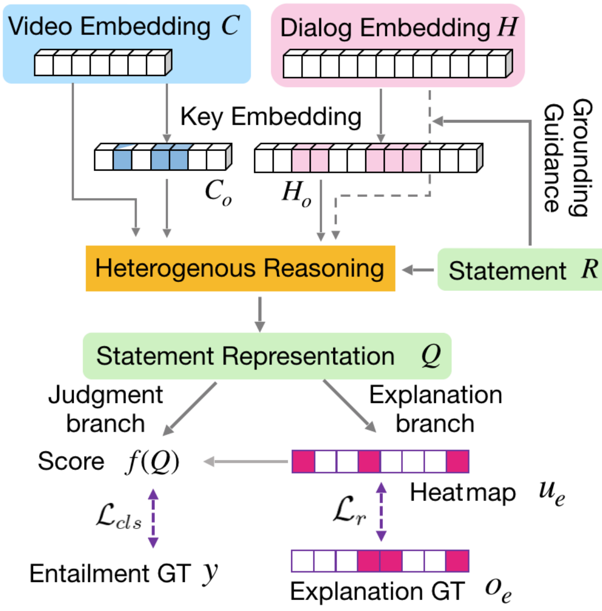


Figure 6.4 Our multi-task learning framework for entailment judgment and its explanation. Given the video and dialog embedding, we use heterogeneous reasoning to fuse them and update the statement representation. Then, the statement representation is incorporated into two branches: the judgment branch to predict if it is entailed or contradicted and the explanation branch to generate a heatmap that shows the contribution of words in the statement in making it fake. GT abbreviates ground-truth.

et al. (2019b):

$$P_* = A_{*\rightarrow s} X_* W_{*s}, \tag{6.5}$$

where $*$ denotes one of the context among video C, dialog D and key C_O, H_O and Adjacency matrix $A_{*\rightarrow s}$ contains the similarity between the statement R and the context embedding X_*. Eq. 6.5 projects the context X_* to an R-shaped embedding P_* by a learnable linear layer W_{*s}. Then, to avoid forgetting, we learn a gating function z_* by a linear operation W_*, b_* and constrained activation $sigmoid$,

$$z_* = sigmoid(W_* \, [R, P_*] + b_*), \tag{6.6}$$

and incorporate the projected embedding P_* of different context into the statement representation by:

$$Q_{*s} = z_* \odot R + (1 - z_*) \odot P_*. \tag{6.7}$$

Eq. 6.7 respectively results in three statement representations Q_{cs}, Q_{hs}, Q_{cos}, Q_{hos} specific to the video, dialog and key context. $\odot$ indicates element-wise product. We concatenate them and update

Method	Visual	Accuracy	Real	Fake	Human-written	Adv-sampled
VIOLIN Liu et al. (2020)	C3D	67.23	74.66	57.73	61.99	67.60
Ours	C3D	68.15	79.21	57.08	61.33	79.43
VIOLIN Liu et al. (2020)	Resnet	67.60	79.10	56.10	59.15	84.49
Ours	Resnet	68.39	79.52	57.25	60.11	84.94

Table 6.1 Entailment Accuracy Comparison. We report the Accuracy (%) of all statements, real statements, fake statements, human-written statements, and adversarially sampled statements. $2/3$ of fake statements are human-written and the remaining $1/3$ are adversarially sampled. Not that "Visual" column denotes the visual features used in the entailment judgment stage.

the statement representation as:

$$Q = [R; Q_{hs}; Q_{cs}; Q_{hos}; Q_{cos}] \,, \tag{6.8}$$

The updated statement representation Q is passed through a function f that contains a linear layer with 1-dimensional output and a sigmoid activation to predict the score of the statement to be real.

6.3.4 Explainable Entailment

The local explanation for judging a textual statement is defined as the contribution of each word, which is in form of a heatmap for a sentence. Our method aims to regularize the training of entailment judgment with its local explanation to promote the model's faithfulness and generalization ability (see the explanation branch in Fig. 6.4) Du et al. (2019). We encourage the entailment model to focus more on the words that actually make the statement contradictory to the video, instead of memorizing the dataset-specific artifacts.

In VIOLIN dataset Lin et al. (2014), more than half of the fake statements were collected by modifying a small subset of the real statement to be contradicted by the video Liu et al. (2020), which makes the difference between the real and fake statements subtle and alleviates the bias. We propose to exploit the subtle difference as a kind of supervision signal for the local explanation. During training, we have access to the real/fake statements that are formed in pairs. For example, a pair of real and fake statements are: "A man in a black jacket gets off his white motorcycle" and "A man in a black jacket gets off the bell towel." respectively. By a simple "*diff*" operation between them, the contradictory items are "the bell towel". The indexes of the different words between the real and fake statements obtained by the "*diff*" operation are defined as the ground-truth of local explanation. We mark it as a binary vector $o_e \in \mathbb{R}^{N_l \times 1}$ that is in length of the statement.

Specifically, we form the entailment judgment (see 3.3) and its explanation as multi-task learning. The explanation branch in Fig. 6.4 takes the updated statement representation Q as input and generates a heatmap $u_e \in \mathbb{R}^{N_l}$ that indicates the contribution of each word to the model prediction $f(Q)$. The explanation loss $\mathcal{L}_r$ is defined as:

$$\mathcal{L}_r = \sum_{i=1}^{N_l} o_e^i(-\log(u_e)) + (1 - o_e^i)(-\log(1 - u_e)), \tag{6.9}$$

which aligns the generated heatmap u_e with the local explanation ground-truth o_e. The overall objective function $\mathcal{L}_e$ is defined as:

$$\mathcal{L}_e = \mathcal{L}_{cls} + \lambda\mathcal{L}_r, \tag{6.10}$$

in which L_{cls} is the binary cross entropy loss for entailment judgment. It balances entailment judgment and its explanation by constraint λ. If a statement is justified as real, each word should be entailed by the premise. Thus, during training, we only regularize the fake statements. During inference, if a statement is predicted as “contradiction”, the explanation module will be triggered to generate the heatmap for the statement.

6.4 Experiments

6.4.1 Dataset

To our best knowledge, Violin Liu et al. (2020) is the only dataset for video entailment task. Violin contains $15,887$ video clips and each video clip is annotated with 3 pairs of real/fake statements, resulting in $95,322$ statements in total. Statements are in random lengths and have 18 words on average. The first two fake statements of each video are human-written by modifying a small portion of the corresponding real statements. Thus, the human-written real/fake statements have very subtle differences, such as one or two words replacement. The third negative statement is adversarially sampled and has a relatively larger difference compared to the real statement. Following the original paper, we split the Violin dataset into 80% for training, 10% for validation, and 10% for testing.

6.4.2 Implementation Details

We use the pre-trained Bert Devlin et al. (2019b) features of both dialog subtitles and statements provided by Liu et al. (2020). For grounding, a Faster R-CNN framework Ren et al. (2015) with VGG-Net Simonyan and Zisserman (2014) as backbone pre-trained on COCO Lin et al. (2014) is applied to extract persons and their features across frames. The entity grounding threshold is set to 0.5. Both the visual and textual input are embedded into d-dimension for fusion, and d is set as 256. We sample the frames corresponding to the middle timestamp of each sentence for grounding. Adam with a learning rate of $1e-3$ is used for optimization. The constraint weight of grounding module β is set to 1. We set batch size as 8 in training. The entities in the statements of other videos in the batch are sampled as the negative samples for training the entity grounding module.

For the contradiction explanation module, we only use the human-written samples for training. Adam with a learning rate of $1e-4$ is used for optimization. Constraint weight of multi-task learning λ is set to 1.

6.4.3 Comparison Methods

We compare our method with the only existing method proposed for the video entailment task, to our best knowledge. Violin Liu et al. (2020) dataset provides a visual/language fusion model to address entailment judgment. The statement representations are jointly modeled with its video and subtitle by an attention-based fusion module.

Experimental results on Violin dataset are shown in Table 6.1. Our proposed explainable entailment model along with grounded evidence given by our method outperforms the previous video entailment method. Because we precisely model the alignment between the video frames and dialog based on grounded evidence. We also evaluate the influence of different visual features following Violin Liu et al. (2020). The results demonstrate that our method works for both image-based features "Resnet" and motion-based features "C3D".

Method	Accuracy	Real Accuracy	Fake Accuracy
v1	66.72	73.60	59.83
v2	67.60	75.50	59.71
v3	66.53	77.78	48.01
Ours	68.39	79.52	57.25

Table 6.2 Ablation Study of Entity Grounding for Entailment (%). **v1**: Removing the first contradiction judgment from the entity grounding module. **v2**: Removing the temporal grounding guidance on entailment judgment. **v3**: Removing $\mathcal{L}_g$.

6.4.4 Ablation Study

6.4.4.1 How does grounding help in entailment?

To exhibit the effectiveness of entity grounding in entailment judgment, we compare our proposed method with the following variants. (1) **v1**: Removing the first contradiction judgment from the entity grounding module. Then, entity grounding is only used to provide temporal guidance. (2) **v2**: Removing the temporal grounding guidance on entailment judgment. We substitute the Eq. 6.8 by $Q = [R; Q_{hs}; Q_{cs}]$. Each frame contributes to the statement without being highlighted. (3) **v3**: Removing $\mathcal{L}_g$. The grounding module is trained without golden negative statements.

Table 6.2 summarizes the results of the aforementioned variants. Comparing "Ours" and **v3**, adding golden negative entities brings more than 1% performance improvement, as it improves the grounding quality. Comparing "Ours" and **v2**, adding temporal grounding's guidance is necessary for making an accurate judgment. The contradiction judgment from the entity grounding module also brings performance gain by comparing "Ours" to **v1**.

6.4.4.2 How does explanation help in entailment?

To explore the contribution of the add-on entailment explanation module, we conduct the ablation study with the following variants: (1) **v4**: Using both the adversarial statements and human-written statements in training the explanation model. (2) **v5**: Removing the explanation regularizer $\mathcal{L}_r$ and only use $\mathcal{L}_{cls}$.

Table 6.3 illustrates the results of the ablation study on the explanation module. The proposed method outperforms the variant **v5** without explanation module by 0.83%, which shows that the multi-task learning boosts the performance of entailment judgment. By the outperformance to the

Method	Accuracy	Real Accuracy	Fake Accuracy
v4	67.65	78.75	56.54
v5	67.32	80.63	54.02
Ours	68.39	79.52	57.25

Table 6.3 Ablation Study of the Add-on Explanation Module for Entailment (%).

Method	Explanation Accuracy
v6	72.42
Ours	75.20

Table 6.4 Quantitative Result for Contradiction Explanation (%).

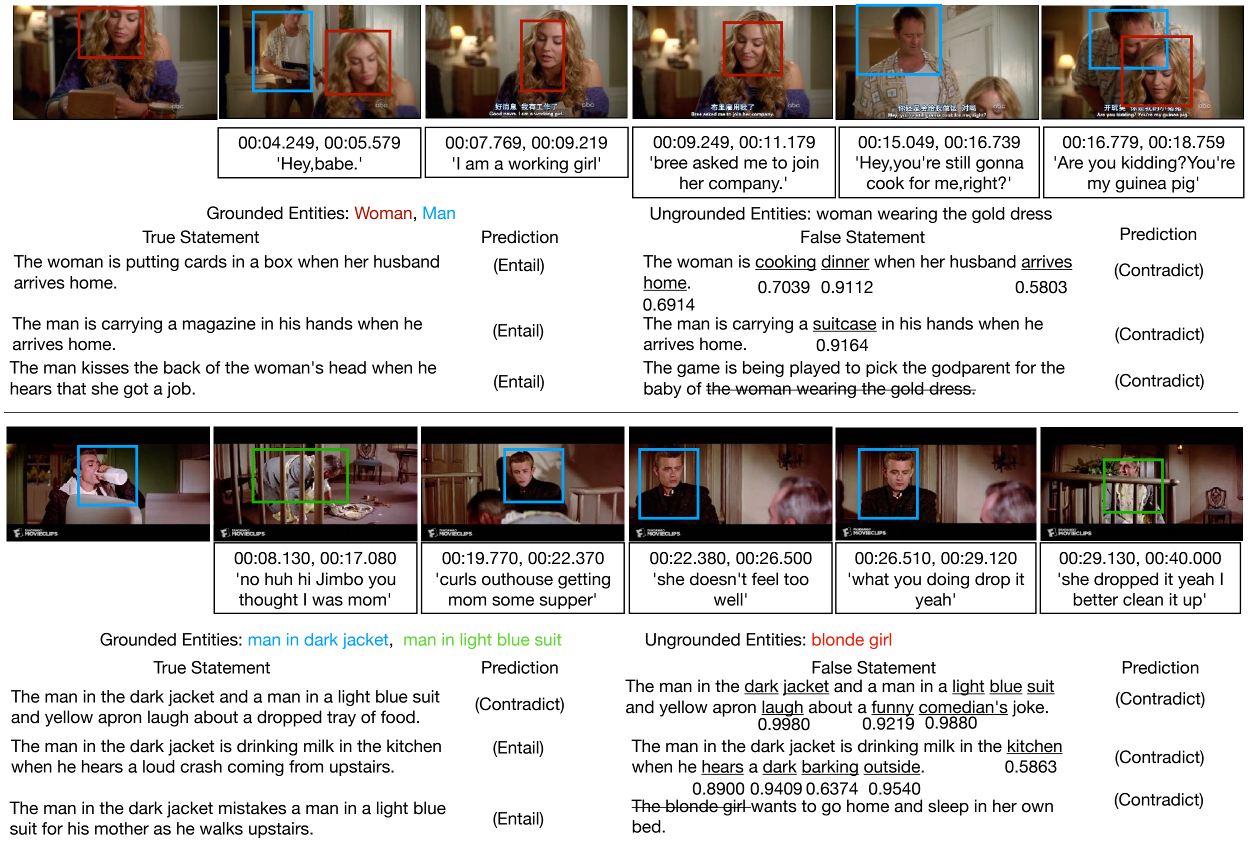
variant **v4**, it is wise to train the explanation model with only human-written samples instead of the adversarial samples, since the adversarial samples are very different from its paired real statement in sentence structure.

6.4.5 Contradiction Explanation Result

Since the real and fake statements are formed in pairs, we can get access to the ground-truth of the items (in words or phrases) that make the statement contradictory to the video. For human-written fake statements, the annotators manually change a small portion of words or phrases in the real statement, which makes the paired real and fake statements have similar grammar and very tiny differences. Thus, the ground-truth of the contradictory items can be obtained by a simple "*diff*" operation between a real/fake pair. But in the adversarial sampled pairs, the real and fake statements are mostly different in structure. Thus, we only use the human-written pairs for training the explanation module. But we test all of the statements either human-written or adversarially sampled.

We quantitatively evaluate the local explanation on the fake statements that are human-written. The evaluation metric is defined as the percentage of the number of words that are correctly explained over the overall number of words in the statement. The explanation results are exhibited in Table 6.4. We achieve 75.2% accuracy in contradiction explanation, which indicates that more than three-quarters of fake words can be found by our explanation model.

We also compare the proposed explanation method with a variant **v6**. **v6** is the variant that explains the entailment of the statement by finding the contradictory constitutes instead of the contradictory words. Constitute parsing method Kitaev and Klein (2018) that was used in obtaining entities in Sec.3.3 is applied to extract constitutes from statement. The result demonstrates that a plain word-level explanation is better than using the constitute.



00:04.249, 00:05.579	00:07.769, 00:09.219	00:09.249, 00:11.179	00:15.049, 00:16.739	00:16.779, 00:18.759
'Hey,babe.'	'I am a working girl'	'bree asked me to join her company.'	'Hey,you're still gonna cook for me,right?'	'Are you kidding?You're my guinea pig'

Grounded Entities: Woman, Man

Ungrounded Entities: woman wearing the gold dress

True Statement	Prediction	False Statement	Prediction
The woman is putting cards in a box when her husband arrives home.	(Entail)	The woman is cooking dinner when her husband arrives home. 0.7039 0.9112 0.5803 0.6914	(Contradict)
The man is carrying a magazine in his hands when he arrives home.	(Entail)	The man is carrying a suitcase in his hands when he arrives home. 0.9164	(Contradict)
The man kisses the back of the woman's head when he hears that she got a job.	(Entail)	The game is being played to pick the godparent for the baby of ~~the woman wearing the gold dress.~~	(Contradict)

00:08.130, 00:17.080	00:19.770, 00:22.370	00:22.380, 00:26.500	00:26.510, 00:29.120	00:29.130, 00:40.000
'no huh hi Jimbo you thought I was mom'	'curls outhouse getting mom some supper'	'she doesn't feel too well'	'what you doing drop it yeah'	'she dropped it yeah I better clean it up'

Grounded Entities: man in dark jacket, man in light blue suit

Ungrounded Entities: blonde girl

True Statement	Prediction	False Statement	Prediction
The man in the dark jacket and a man in a light blue suit and yellow apron laugh about a dropped tray of food.	(Contradict)	The man in the dark jacket and a man in a light blue suit and yellow apron laugh about a funny comedian's joke. 0.9980 0.9219 0.9880	(Contradict)
The man in the dark jacket is drinking milk in the kitchen when he hears a loud crash coming from upstairs.	(Entail)	The man in the dark jacket is drinking milk in the kitchen when he hears a dark barking outside. 0.5863 0.8900 0.9409 0.6374 0.9540	(Contradict)
The man in the dark jacket mistakes a man in a light blue suit for his mother as he walks upstairs.	(Entail)	~~The blonde girl~~ wants to go home and sleep in her own bed.	(Contradict)

Figure 6.5 Visualization of the entailment judgment and its explanation with grounded evidence. Strikethrough indicates that the video does not contain the described entity and thus is judged as "contradiction". The contradictory items are marked by the underline with the predicted scores.

6.4.6 Explainable Entailment Result

Fig. 6.5 presents several entailment judgment examples using our method. Our model can successfully ground the described entities to the specific regions and the relevant frames, even if the grounding annotation is not provided in training. Our model also has the resilience to the entities in the fake statements that are absent in the video. The two fake statements contain the entities that are missing (*e.g.*, , "the blonde girl", "the woman wearing the golden dress"), marked by strikethrough, and are judged as fake in the grounding stage. The predicted fake items are marked by the underline with explanation scores. We find if the statement is correctly judged as fake, the explanation result is more reliable.

6.5 Summary

In this paper, we present a novel approach for video entailment and its local explanation. Entity grounding is highly incorporated into our task from two aspects. First, we train a weakly-supervised entity video grounding module to judge a statement as "contradiction" if the statement consists of an entity absent in the video. Then if the entity is present in the video, we infer the temporal occurrence of that entity to guide the entailment judgment module focusing on the entity-relevant clips. In addition to entailment judgment, our method is also developed to explain which words or phrases make the statement contradictory to the video. We formulate the local explanation as a regularizer to the decision-making of entailment to improve the model's faithfulness. Extensive results on VIOLIN dataset demonstrate the resulting model consistently outperforms the existing methods.

BIBLIOGRAPHY

Antol, S., Agrawal, A., Lu, J., Mitchell, M., Batra, D., Lawrence Zitnick, C., and Parikh, D. (2015). Vqa: Visual question answering. In *CVPR*, pages 2425–2433.

Bowman, S. R., Angeli, G., Potts, C., and Manning, C. D. (2015). A large annotated corpus for learning natural language inference. In *EMNLP*.

Camburu, O.-M., Rocktäschel, T., Lukasiewicz, T., and Blunsom, P. (2018). e-snli: Natural language inference with natural language explanations. In *NIPS*.

Chen, J., Bao, W., and Kong, Y. (2020). Activity-driven weakly-supervised spatio-temporal grounding from untrimmed videos. In *ACM Multimedia*, pages 3789–3797.

Chen, K., Gao, J., and Nevatia, R. (2018). Knowledge aided consistency for weakly supervised phrase grounding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4042–4050.

Chen, L., Zhai, M., He, J., and Mori, G. (2019a). Object grounding via iterative context reasoning. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 0–0.

Chen, Z., Ma, L., Luo, W., and Wong, K.-Y. K. (2019b). Weakly-supervised spatio-temporally grounding natural sentence in video.

Condoravdi, C., Crouch, D., De Paiva, V., Stolle, R., and Bobrow, D. (2003). Entailment, intensionality and text understanding. In *NAACL*.

Dagan, I., Glickman, O., and Magnini, B. (2005). The pascal recognising textual entailment challenge. In *Machine Learning Challenges Workshop*. Springer.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019a). BERT: Pre-training of deep bidirectional transformers for language understanding.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019b). Bert: Pre-training of deep bidirectional transformers for language understanding. In *ACL*, pages 4171–4186.

Do, V., Camburu, O.-M., Akata, Z., and Lukasiewicz, T. (2020). e-snli-ve-2.0: Corrected visual-textual entailment with natural language explanations. *arXiv preprint arXiv:2004.03744*.

Du, M., Liu, N., Yang, F., and Hu, X. (2019). Learning credible deep neural networks with rationale regularization. In *ICDM*.

Huang, D.-A., Buch, S., Dery, L., Garg, A., Fei-Fei, L., and Niebles, J. C. (2018). Finding "it": Weakly-supervised, reference-aware visual grounding in instructional videos. In *IEEE*

Conference on Computer Vision and Pattern Recognition (CVPR).

Karpathy, A. and Fei-Fei, L. (2015). Deep visual-semantic alignments for generating image descriptions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3128–3137.

Kim, J., Ma, M., Kim, K., Kim, S., and Yoo, C. D. (2019). Progressive attention memory network for movie story question answering. In *CVPR*.

Kim, J., Ma, M., Pham, T., Kim, K., and Yoo, C. D. (2020). Modality shifting attention network for multi-modal video question answering. In *CVPR*.

Kim, K.-M., Choi, S.-H., Kim, J.-H., and Zhang, B.-T. (2018). Multimodal dual attention memory for video story question answering. In *ECCV*.

Kitaev, N. and Klein, D. (2018). Constituency parsing with a self-attentive encoder. In *ACL*, pages 2676–2686.

Lei, J., Yu, L., Bansal, M., and Berg, T. L. (2018). Tvqa: Localized, compositional video question answering.

Lei, J., Yu, L., Berg, T. L., and Bansal, M. (2019). Tvqa+: Spatio-temporal grounding for video question answering.

Lei, J., Yu, L., Berg, T. L., and Bansal, M. (2020). Tvr: A large-scale dataset for video-subtitle moment retrieval. *arXiv preprint arXiv:2001.09099*.

Li, K., Zhang, Y., Li, K., Li, Y., and Fu, Y. (2019). Visual semantic reasoning for image-text matching. In *ICCV*, pages 4654–4662.

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. (2014). Microsoft coco: Common objects in context. In *ECCV*.

Liu, J., Chen, W., Cheng, Y., Gan, Z., Yu, L., Yang, Y., and Liu, J. (2020). Violin: A large-scale dataset for video-and-language inference. In *CVPR*.

MacCartney, B. and Manning, C. D. (2009). An extended model of natural logic. In *Proceedings of the Eight International Conference on Computational Semantics*.

Na, S., Lee, S., Kim, J., and Kim, G. (2017). A read-write memory network for movie story understanding. In *ICCV*.

Ren, S., He, K., Girshick, R., and Sun, J. (2015). Faster r-cnn: Towards real-time object detection with region proposal networks. In *NIPS*.

Rohrbach, A., Rohrbach, M., Hu, R., Darrell, T., and Schiele, B. (2016). Grounding of textual phrases in images by reconstruction. In *European Conference on Computer Vision*, pages 817–834. Springer.

Shi, J., Xu, J., Gong, B., and Xu, C. (2019). Not all frames are equal: Weakly-supervised video grounding with contextual similarity and visual clustering losses. In *CVPR*, pages 10444–10452.

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

Suhr, A., Zhou, S., Zhang, A., Zhang, I., Bai, H., and Artzi, Y. (2019). A corpus for reasoning about natural language grounded in photographs. In *ACL*.

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

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

Xie, N., Lai, F., Doran, D., and Kadav, A. (2019). Visual entailment: A novel task for fine-grained image understanding. *arXiv preprint arXiv:1901.06706*.

Young, P., Lai, A., Hodosh, M., and Hockenmaier, J. (2014). From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *TACL*.

Zhang, C., Song, D., Huang, C., Swami, A., and Chawla, N. V. (2019). Heterogeneous graph neural network. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*.

Zhou, L., Kalantidis, Y., Chen, X., Corso, J. J., and Rohrbach, M. (2019). Grounded video description. In *CVPR*.

Zhou, L., Louis, N., and Corso, J. J. (2018). Weakly-supervised video object grounding from text by loss weighting and object interaction.

Zhou, Y., Wang, M., Liu, D., Hu, Z., and Zhang, H. (2020). More grounded image captioning by distilling image-text matching model. In *CVPR*.

CHAPTER 7

CONCLUSIONS AND FUTURE WORK

7.1 Summary of the thesis

In this thesis, we have posed and investigated a range of fundamental problems in long-form video understanding. We developed a series of learning frameworks that address representation learning, temporal dependency modeling, and trustworthy video understanding. Through extensive investigations, we can draw the following conclusions.

First, we propose a novel framework to solve the VideoQA. Our method addresses the importance of temporality reasoning. To this end, we realize that it is worth revisiting optical flow, as flow may become less considered in atomic action recognition but is still effective in long-horizon temporality. Then, we propose action-centric contrastive learning that makes both video and text representations informative for action. Then, we fine-tune the VideoQA via a novel temporal sensitivity-aware confusion loss to mitigate the potential static bias. Our ATM method is demonstrated to be superior to all existing VideoQA methods on multiple benchmarks and shows faithful temporality reasoning via a new metric.

Second, we have proposed a novel sequential relational anticipation model (SRAM) to predict group activity given only the beginning frames of an activity execution. Our model captures the complex relational dynamics of multiple people in the observed frames. It then anticipates the group representations including group activity features and position features. A novel sequential decoder is proposed to progressively anticipate the group representations through several unrolling stages. Extensive results on two datasets demonstrate that our method significantly outperforms the state-of-the-art methods. Results also validate that the progressive anticipation using multiple unrolling stages facilitates group activity prediction. Further experimental results show that the modeling and prediction of people's positions improve our performance on group activity prediction.

Third, we present GateHUB for online action detection in untrimmed streaming videos. It consists of novel designs including Gated History Unit (GHU), Future-augmented History (FaH), and a background suppression loss to more informatively leverage history and reduce false positives for current frame prediction. GateHUB achieves higher accuracy than all existing methods for online action detection and is more efficient than the existing best method. Moreover, its optical flow-free variant is 2.8× faster than previous methods that require both RGB and optical flow while obtaining higher or close accuracy.

Fourth, we investigate the spatio-temporal grounding in untrimmed videos with frequent visual inconsistency in a weakly-supervised manner. We develop two novel MIL ranking losses for the spatial and temporal domains. Furthermore, to bridge the granularity gap between the coarse text information and the detailed visual information, we introduce an activity-driven object state encoding module to enhance textual representation. Experiments on two popular datasets demonstrate the superiority of our method and its generalization ability to other datasets with unseen queries.

Fifth, we present a novel approach for video entailment and its local explanation. Entity grounding is highly incorporated into our task from two aspects. First, we train a weakly-supervised entity video grounding module to judge a statement as "contradiction" if the statement consists of an entity absent in the video. Then if the entity is present in the video, we infer the temporal occurrence of that entity to guide the entailment judgment module focusing on the entity-relevant clips. In addition to entailment judgment, our method is also developed to explain which words or phrases make the statement contradictory to the video. We formulate the local explanation as a regularizer to the decision-making of entailment to improve the model's faithfulness. Extensive results on VIOLIN dataset demonstrate the resulting model consistently outperforms the existing methods.

7.2 Future work

7.2.1 Leveraging temporal reasoning for social good

In addition to videos, the temporal structure is also present in many specialized domains. For example, in biomedicine, doctors can infer "A tumor is decreasing under chemotherapy treatment", given the comparison of a prior and a current radiology image. I am interested in enhancing downstream applications such as report generation to leverage temporal structure. Broadly speaking, I am committed to grounding this technique in a context of social good, such as applications in healthcare, education, social inclusion, and biomedicine. I am passionate about working with frontline researchers in the medical field, computational social science, art, and data mining to improve these applications by integrating temporal semantic prior, as well as learning in a self-supervised manner with complementary temporal signals.

7.2.2 Human-AI interactive systems to perceive accessibility

I believe video understanding has a natural benefit in assisting people with disabilities. A significant case is to adapt computer vision techniques to overcome the visual challenge for blind people so that a blind person can know the surrounding physical world by wearing a camera with on-device computation. Deaf people use visual language as a means of communication, which could be largely benefited by video understanding techniques. Achieving automatic sign language recognition and localization enables the construction of sign language dictionaries and real-time communication with surroundings. To this end, I will investigate video+language techniques to progress in the availability of these applications. I am excited to work with researchers in Human-Computer Interaction and NGO to understand the real challenge of people from neglected groups.

7.2.3 Holistic trustworthy computer vision

In the era of multi-modal foundation models and large language models, my vision is to attain holistic trustworthiness in machine learning models. In addition to bias mitigation and explainability that we have studied, trustworthiness includes privacy, inclusiveness, security, fairness, and other core targets. For example, recognizing human action requires the model to preserve the privacy of sensitive sectors such as gait and identity in training data, be inclusive to different data nodes, and make fair decisions. Although videos constitute a substantial portion of computer vision data, developing trustworthy video models are much more underscored than image or graph data counterpart. I plan to train researchers thinking about the risks and challenges of real-world AI systems. I want to collaborate with people from cybersecurity and industry in these domains.