

Title	Feature Representations for Visual and Language Task: Towards Deeper Video Understanding
Author(s)	楊, 沢坤
Citation	大阪大学, 2021, 博士論文
Version Type	VoR
URL	https://doi.org/10.18910/85435
rights	© The Author(s) 2023. This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/ .
Note	

Feature Representations for Visual and Language
Task: Towards Deeper Video Understanding

Submitted to
Graduate School of Information Science and Technology
Osaka University

June 2021

YANG Zekun

Thesis Committee

Prof. Haruo Takemura (Osaka University)

Prof. Yasushi Yagi (Osaka University)

Prof. Noriyuki Miura (Osaka University)

Associate Prof. Yuta Nakashima (Osaka University)

List of Publications

Peer-Reviewed Journal Articles

1. Yang, Z., Garcia, N., Chu, C., Otani, M., Nakashima, Y., Takemura, H. (2021). A Comparative Study of Language Transformers for Video Question Answering. *Neurocomputing*. (Chapter 4)

Peer-Reviewed Conference Proceedings

1. Yang, Z., Garcia, N., Chu, C., Otani, M., Nakashima, Y., Takemura, H. (2020). Bert representations for video question answering. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision* (pp. 1556-1565). (Chapter 3)

Non-Peer-Reviewed Conference Proceedings

1. Yang, Z., Nakashima, Y., Takemura, H. (2021). Predicting Humor in Visual and Language Modalities. In *Technical Committee on Pattern Recognition and Media Understanding (PRMU)* (pp. 1-6). (Chapter 5)

Abstract

This research is an attempt to promote deeper understanding of videos. It mainly focuses on feature representations for visual and language elements, and carries the research in three aspects based on two different tasks.

First, it focuses on video question answering task, which aims at answering questions about a video. Currently, most work focuses on image-based question answering, and less attention has been paid to answering questions about videos. However, video question answering presents some unique challenges that are worth studying: It not only requires modelling a sequence of visual features over time, but also needs to reason about associated subtitles. Hence, BERT, a sequential modelling technique based on Transformers, is used to encode the complex semantics from video clips. The proposed model jointly captures the visual and language information of a video scene by encoding not only the subtitles but also a sequence of visual concepts with pre-trained BERT model. In the experiments, the performance of the proposed model is studied by taking different input arrangements, showing outstanding improvements when compared against previous work on two video question answering datasets: TVQA and Pororo.

Then, a deeper study about Transformers is carried. Transformer is a novel architecture that aims at solving sequence-to-sequence tasks while handling long-range dependencies. It relies on self-attention to compute representations of the input and output without using RNNs. In recent years, many kinds of Transformers have been proposed and used extensively. They have been reported to outperform RNNs in several natural language processing tasks. Before carrying on this study, it is not clear which Transformer performs the best in video question answering task, and why it performs the best. With the aim of understanding Transformer better, the visual semantics and the subtitles are encoded with four commonly applied Transformers: BERT, XLNet, RoBERTa, and ALBERT. It is found that the accuracy is different when different Transformers are adopted on the same dataset, and the reason behind such differences is related to the different pre-training settings in different Transformers.

Finally, this research focuses on multi-modal humor prediction task. Humor provokes laughter and provides amusement, and can be induced by signals in the visual, linguistic, and vocal modalities. Previous methods mainly predict humor in the sentence level and with single modality, which often ignore humor caused by, for example, actions. In this work, a dataset for multi-modal humor prediction is proposed based on the famous sitcom *the Big Bang Theory*. Next, a method is introduced to find temporal segments that involve humor in videos. This method adopts a sliding window to divide the video, and model the visual modality described by pose and facial features, along with the linguistic modality given as subtitles to predict humor. Experimental results show that our method helps improve the performance of humor prediction.

Keywords: Video Understanding, Visual and Language Task, Feature Representation

Acknowledgements

First of all, I would like to thank Osaka University for giving me a chance to study abroad. Before coming to Osaka University, I was wondering whether I could get used to this unfamiliar environment and I even thought of giving up. After coming to Osaka University, I was exempted from most of tuition fees and could study here with very few burdens. I have been nourished with the abundant resources and excellent environments here and can stick to a higher goal during these years.

Second, I would like to thank all the teachers in Takemura laboratory and institution of datability science. They are friendly but strict to us. When we need help, they often come forward and give assistance to us. When we meet difficulties, they always encourage us and help us with their best knowledge. When we make mistakes in research, they give strict critique and point out the problems directly. In all, I would like to express my sincere gratitude to my teachers.

I would also like to thank my colleagues and my friend, like Takaya Ogawa, Kazuki Ogawa, Azusa Nagahama, Yusuke Takayama, Harn Sison, Tao Tao, Guanghan Zhao, Naruhiro Yamane, Koshiro Motoki, and Tianxu Han. I can communicate with them easily and obtain their valuable comments during these years.

At last, I would like to thank the Chinese government for providing most of my funds since Sep. 2019 and offering special health kits during the COVID-19 period.

Contents

1	Introduction	1
1.1	The Background of the Research	1
1.2	Visual and Language Tasks	2
1.3	Feature Representation in Visual and Language Tasks	4
1.4	Video Understanding	5
1.5	The Challenges in Video Feature Representations	7
1.6	Goal, Task, and Contribution	8
1.7	The Structure of the Thesis	9
2	Related Work	12
2.1	Visual Question Answering Task	12
2.1.1	VQA Dataset	13
2.1.2	VQA Method	15
2.2	Video Question Answering Task	16
2.2.1	Video-QA Datasets	17
2.2.2	Video-QA Methods	18
2.3	Emotion Prediction Task	19
2.3.1	Emotion Prediction Datasets	20
2.3.2	Emotion Prediction Methods	21
2.4	Feature Representation Methods	22
3	BERT Representations for Video Question Answering	24
3.1	Introduction	24
3.2	Prior Work	26
3.2.1	Video Question Answering	26

3.2.2	Language Representation	27
3.3	The Proposed Method	28
3.3.1	Introduction to BERT	29
3.3.2	Feature Representations	30
3.3.3	Answer Prediction	32
3.4	Experimental Results	33
3.4.1	Experimental Settings	33
3.4.2	Datasets	33
3.4.3	Input Sequences	35
3.4.4	Results on the TVQA Dataset	35
3.4.5	Results on the Pororo Dataset	36
3.4.6	Ablation Study	37
3.5	Discussion	38
3.5.1	Training Time	38
3.5.2	Evaluation with Full-Length Elements	38
3.5.3	Evaluation with Different Sequence Lengths	41
3.6	Qualitative Results	42
3.7	Summary	44
4	A Comparative Study of Language Transformers for Video Question Answering	47
4.1	Introduction	47
4.2	Prior Work	50
4.2.1	Language Transformers	50
4.2.2	The Applications of Language Transformers	51
4.3	Introduction to Transformers	51
4.3.1	BERT	52
4.3.2	XLNet	53
4.3.3	RoBERTa	53
4.3.4	ALBERT	54
4.4	The Proposed Method	54
4.4.1	Visual Representations	55

4.4.2	Language Representations	56
4.4.3	Prediction	57
4.5	Experiments	57
4.5.1	Experimental Setting	58
4.5.2	Results on the TVQA Dataset	59
4.5.3	Results on the Pororo Dataset	64
4.6	Discussion	66
4.6.1	Training Time and Inferring Time	66
4.6.2	Comparison with High-level Visual Features	67
4.6.3	Attention Weights in Different Flows	68
4.6.4	Evaluation with Different Sequence Lengths	70
4.6.5	Qualitative Results	70
4.7	Summary	73
5	Multi-modal Humor Segment Prediction in Video	75
5.1	Introduction	75
5.2	Prior Work	78
5.2.1	Humor Prediction Methods	78
5.2.2	Humor Generation Methods	79
5.3	The Proposed Datasets	79
5.4	The Proposed Model	81
5.4.1	Sliding Window	82
5.4.2	Pose Flow	83
5.4.3	Face Flow	84
5.4.4	Language Flow	85
5.4.5	Prediction and Training	86
5.4.6	Converting Frame Predictions to Temporal Segments	86
5.5	Experimental Results	87
5.5.1	Quantitative Results	87
5.5.2	Qualitative Results	91
5.6	Discussion	93
5.6.1	Training Time	93

5.6.2	Training with All the Joints in 3D Pose	93
5.6.3	Training with Different Lengths of Sliding Window . . .	95
5.7	Summary	96
6	Conclusion and Outlook	98
6.1	Conclusion	98
6.2	Outlook	99
	References	102

List of Tables

3.1	The training parameters in the experiment.	33
3.2	Accuracy (in %) of the proposed method on TVQA dataset with time stamp annotations. Note that one only has limited chances to submit their results to the test server for evaluation, thus only some representative results of the proposed method are shown to public.	36
3.3	Accuracy (in %) of the proposed method on Pororo dataset. . . .	37
3.4	Training time of one epoch in TVQA dataset (h:mm)	38
3.5	Accuracy results (in %) of the proposed method on the TVQA dataset using full-length subtitles (without timestamps annotations). Note that one only has limited chances to submit their results to the test server for evaluation, thus only some representative results of the proposed method are shown to public. . .	39
3.6	Statistics of the input sequences on the TVQA test* set.	41
3.7	Accuracy results (in %) on the TVQA test* set using different L values (128, 256, and 512).	42
4.1	The training parameters in the experiment.	58
4.2	Detailed information about different pre-trained Transformer models.	58
4.3	Accuracy (in %) of proposed model and baselines on TVQA dataset with time stamp annotations. Note that one only has limited chances to submit their results to the test server for evaluation; thus, only some representative results of the proposed model are shown to the public.	60

4.4	Accuracy (in %) of proposed model and baselines using full length elements on TVQA dataset.	63
4.5	Accuracy (in %) of proposed model and baselines on Pororo dataset.	65
4.6	Training time (in h:mm, per epoch) and inferring time (in ms, per question) for different Transformers on the TVQA dataset. .	67
4.7	Accuracy of question answering using different features (in %). .	68
4.8	Accuracy results (in %) on the TVQA test* set using different L values (128, 256, and 512).	70
5.1	Statistics of the proposed dataset.	80
5.2	The training parameters in the experiment.	87
5.3	Frame level results on <i>the Big Bang Theory</i> test set (in %) . . .	88
5.4	Segment level results on <i>the Big Bang Theory</i> test set (in %) . .	89
5.5	Frame level results of Kayatani et al's method and the proposed method (in %)	89
5.6	Segment level results of Kayatani et al's method and the proposed method (in %)	90
5.7	Training time on <i>the Big Bang Theory</i> dataset per epoch (in mm:ss)	93
5.8	Frame level results on <i>the Big Bang Theory</i> test set with all joints in 3D pose (in %)	94
5.9	Segment level results on <i>the Big Bang Theory</i> test set with all joints in 3D pose (in %)	94
5.10	Frame-level results on <i>the Big Bang Theory</i> test set under different lengths (in %)	95
5.11	Segment-level results on <i>the Big Bang Theory</i> test set under different lengths (in %)	96

List of Figures

1.1	Different types of visual and language task.	2
1.2	Typical applications of visual and language task.	3
1.3	The tasks handle videos that contain a set of frames and relatively long subtitles.	6
2.1	A basic flow chart of VQA system.	13
2.2	An example of question answering in video	17
2.3	A typical concept of emotion prediction task in multiple modalities	20
3.1	The proposed model for video-QA based on BERT.	26
3.2	The framework of the proposed method.	28
3.3	An example of input representation using BERT.	29
3.4	figure.caption.27	
3.5	Example of the TVQA dataset.	34
3.6	Example of the Pororo dataset.	34
3.7	Successful and unsuccessful predictions of the proposed model and TVQA model. Note that the bounding boxes or words in blue are hints to the answer, which are annotated manually. . .	43
4.1	Overview of the research. Variants of Transformers are used to encode the complex semantics from video clips in video question answering task.	48
4.2	RNNs that encode the input sentence word by word, while Transformers treat the sentence as a whole.	52

4.3	Proposed model for answering multiple choice video questions. Note that V, Q, and S denote visual concept, question, and subtitle, respectively.	55
4.4	Examples of detected features in TVQA and Pororo datasets. The top line shows frames from the TVQA dataset, and the bottom line shows frames from the Pororo dataset.	66
4.5	Attention weights of the BERT model in the visual and language flow. (a) Visual flow, Layer 1, (b) Visual flow, Layer 7, (c) Visual flow, Layer 12, (d) Language flow, Layer 1, (e) Language flow, Layer 7, (f) Language flow, Layer 12. Note that the height of the full attention visualization is very large, so the figure only puts the focus (i.e. where the attention focuses) here.	69
4.6	Successful and unsuccessful predictions on TVQA and Pororo datasets. (1) to (4) are examples from the TVQA dataset, (5) and (6) are examples from the Pororo dataset. Note that the bounding boxes or words in blue are hints to the answer, which are annotated by us manually.	72
5.1	Some examples of humor.	76
5.2	Overview of the task. This work is going to predict humor segments in the video.	77
5.3	The proposed dataset for humor prediction, which contains video frames, subtitles, and wave envelopes.	80
5.4	The ground-truth humor segments can be obtained by setting up threshold and binarizing the wave envelope. The boxes in orange are wave envelopes with noise, and the boxes in green are the fixed humor segments.	81
5.5	The proposed model for multi-modal humor prediction.	82
5.6	Examples of original video frame and pose detection. (a) Video frames. (b) 2D pose. (c) 3D pose	83
5.7	Examples of original video frame and face detection. (a) Video frames. (b) Face landmark. (c) Action unit.	85

5.8	Examples of humor predictions using the proposed method and Kayatani et al's method.	92
-----	---	----

Chapter 1

Introduction

This chapter gives a brief introduction to the background of the whole research. Including visual and language task, feature representation and its challenges, the general goal, selected task and contribution, and the structure of the thesis.

1.1 The Background of the Research

Artificial intelligence (AI) is the simulation of human intelligence in machine systems (McCarthy, 2007). An AI system is programmed to behave like humans and mimic their actions. The basic theory of AI was first proposed in 1956. During the past 65 years, it has been quickly developing to meet various social demands. Now, AI has become a frontier technology with wide applications and blended backgrounds. In recent years, AI technique have been embedded in multiple ways. Lots of related tasks have been proposed to broaden its applications. In the future, AI technique will keep developing to help the computer understand the world in a more comprehensive manner and communicate with people in a more intelligent way. The application of AI will also lead to an intelligent trend in both industries and our daily life.

Every day, we use various signals for communications, including language, sound, image, gesture, etc. Among these signals, visual and language are the basis of our thinking and value. They work as essential tools for us to learn the world and express ourselves in our daily life. We would also want the AI system to know what we see and how we think so that it can understand the visual and language signals made by us and better respond to our requirements. Based on



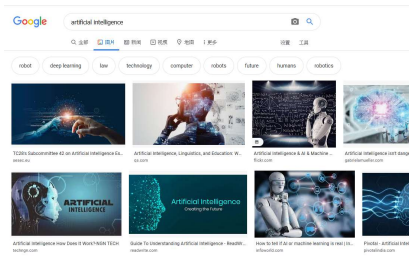
Figure 1.1. Different types of visual and language task.

this idea, it is important to take a deep study on AI systems that combine visual and language signals together.

1.2 Visual and Language Tasks

Visual and language task processes both visual and language information in computers. It usually combines the inputs with visual and language features together and uses AI-based methods to generate a prediction as its output (Anderson et al., 2018; Lei, Yu, Bansal, & Berg, 2018; Das, Kottur, Gupta, et al., 2017). Generally speaking, the goal of such tasks is to make the computer better understand what is happening based on the inputs. Thus, the computer needs a comprehensive understanding of both visual and language features to handle such kind of tasks.

Visual and language tasks have many branches. Some instances of different types of tasks are shown in Fig. 1.1. Different types of tasks might have different goals. Improving their performance will give better predictions and improve



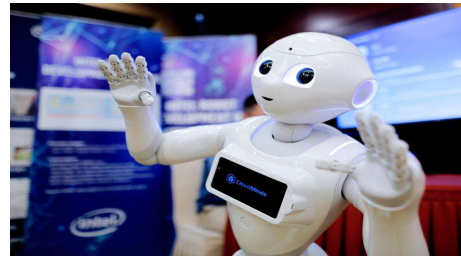
(a) Image Search Engine



(b) Blind Assistance



(c) Human-machine Interaction



(d) Robotics

Figure 1.2. Typical applications of visual and language task.

the efficiency of AI system. For example, in visual dialogue tasks (Das, Kottur, Gupta, et al., 2017; Das, Kottur, Moura, Lee, & Batra, 2017) (Fig. 1.1 (a)), the computer describes the visual content comprehensively and holds a meaningful dialog with humans in a natural and conversational way, in image captioning tasks (Gan et al., 2017; Yao, Pan, Li, & Mei, 2018) (Fig. 1.1 (b)), the computer generates captions correctly for different kinds of images in the large database, in visual question answering tasks (H. Xu & Saenko, 2016; H. Liu et al., 2018) (Fig. 1.1 (c)), the computer answers questions related to the images automatically, and in video moment retrieval tasks (M. Liu et al., 2018; Lei, Yu, Berg, & Bansal, 2020a) (Fig. 1.1 (d)), the computer needs to find the related video moments based on the query given by human.

On the other hand, visual and language task has a wide range of applications, like some instances in Fig. 1.2. These applications propose different requirements to the task. The improvements in visual and language tasks broaden the application of new techniques and give better experiences to the users. For example: The image search engine (Fig. 1.2 (a)) takes some language keywords

as input and finds a number of related images from the corpus as output; blind assistance system (Fig. 1.2 (b)) processes various questions about the outside environment and gives response to its users in a brief but accurate way; human-machine interaction task (Fig. 1.2 (c)) recognizes both the neighboring environment and the words from the user and helps the system interact with users; and robot (Fig. 1.2 (d)) understands the commands and makes efficient solutions based on its surroundings.

Visual and language tasks make predictions by making full use of the information in both visual and language modalities. The input of these tasks usually consists of two flows: One flow processes image to obtain visual features and the other flow processes text to obtain language features. Then, features in both inputs are represented by high-dimensional vectors that contain deep semantic information. Neural networks are adopted to learn the connections between the high-dimensional representations and the possible prediction in each flow. As the visual features and the language features complement each other, they are usually combined together to give joint information for the final prediction. However, present visual and language tasks still face some challenges, leaving much space for potential work to improve their performance. This work would like to select some visual and language tasks and improve them by finding appropriate feature representations for their input elements. By doing so, the intelligent system can have a deeper understand the visual and language elements and make better predictions.

1.3 Feature Representation in Visual and Language Tasks

Feature representation uses certain methods to convert the input features into high-level vectors that contain deep semantic information. For visual and language tasks, the features in visual and language inputs are often obtained in different ways. Lots of work (K. He, Zhang, Ren, & Sun, 2016a; W. Liu et al., 2016; Anderson et al., 2018; Ben-Younes, Cadene, Cord, & Thome, 2017) retrieves visual features by setting up deep neural networks that usually pre-trained on large-scale image datasets (e.g. Visual Genome (Krishna, Hata, Ren,

Fei-Fei, & Carlos Niebles, 2017), ImageNet (Deng et al., 2009) and Microsoft COCO (Lin et al., 2014)) to detect objects in the images. The language features are often retrieved based on words or sentences in the texts. Next, certain mapping is created to connect them with high-dimensional, real-valued vectors, and convert them to the form that can be processed in the computer. Afterwards, these vectors are fed into neural networks. The networks are then being trained to get the final prediction. In the real case, the network is usually expected to be trained on a large-scale dataset to reduce over-fitting. Since it is time-consuming to train on a large-scale dataset from scratch, some pre-trained models are taken in the training part of the network. To let these models work on the corresponding tasks, fine-tuning is usually needed to adjust the parameters of these models.

Feature representation plays one of the vital roles in visual and language tasks. It not only converts the retrieved visual and/or language features into high-dimensional vectors that can be understood by computers, but also builds a bridge between human language and computer language. Finding appropriate feature representations reduces the gap between human and machine, and conveys the information in the visual and language inputs more comprehensively. Thus, a better feature representation is helpful to improve the performance of the task.

1.4 Video Understanding

Video is an important medium in our daily life. It has moving visual and audio media and contains rich information. In this work, video is selected as the source of the research because it contains visual and language features from the real world, which are materials that many intelligent systems need to handle. By carrying research on videos, the intelligent system will know the world better and become more familiar with the way that human thinks. Based on this idea, the proposed research handles videos that contain a set of frames and relatively long subtitles, which is shown in Fig. 1.3.

To understand the video, the computer needs to recognize and localize certain

Video Frames:**Subtitles:**

You will answer every page at a run. A run!

Bailey: That's rule number two.

Bailey: Your first shift starts now and lasts 48 hours. You're interns, grunts, nobodies.

Bailey: bottom of the surgical food chain.

Figure 1.3. The tasks handle videos that contain a set of frames and relatively long subtitles.

events inside it. Furthermore, some predictions may need to be made based on them. Thus, the computer needs to know what is happening inside the video by fully retrieving the features and understanding the contents.

There are some typical instances for video understanding. For example, video classification (Abu-El-Haija et al., 2016) aims at producing a related label according to its frames; video anomaly detection (Patil & Biswas, 2016) tries to detect the abnormal events that appeared in the video; video captioning (J. Wang, Jiang, Ma, Liu, & Xu, 2018) is the task to generate several sentences and ground them automatically with time locations, and video question answering (Lei et al., 2018) is the task to let the system to retrieve video features and answer questions correctly.

Since the video contains a series of frames that connect with each other in temporal domains, it is highly challenging for an intelligent system to understand the video in a deeper manner. To promote deeper video understanding, making use of the temporal information between video frames is an essential way. The temporal information demonstrates the evolution of the video plots and provides important information to recognize and localize the events in the video. By making use of the temporal information, the computer can know what is happening in the video, and is able to make some predictions based on them.

1.5 The Challenges in Video Feature Representations

Feature representation has become the key to understand the video in a deeper manner, and is one of the important bases to make a good prediction based on the video contents. Recently, lots of work (Anderson et al., 2018; Lei et al., 2018; Lei, Yu, Berg, & Bansal, 2019; H. Xu & Saenko, 2016; Gokhale, Banerjee, Baral, & Yang, 2020; Xi, Zhang, Ding, & Wan, 2020) has been proposed to explore feature representations of visual and language inputs in different aspects.

On the other hand, however, finding appropriate representations in the video is a very challenging problem because of the following reasons:

First, existing methods tend to represent the image with high-level features. However, some evidence has proved (Lei et al., 2018) that high-level visual features cannot perform very well on visual and language tasks. Since different visual and language tasks have different goals, it is necessary to find more appropriate features to improve the performance of the visual and language task by considering the characteristic of each task.

Second, the elements like video frames and the texts contain abundant information that might be helpful to the task. This kind of information is intuitive to human, but is not easy for computers to process. Previous work (Lei et al., 2018; Ogawa, Sasaka, Maeda, & Haseyama, 2018) tends to use long short-term memory (LSTM) (Hochreiter & Schmidhuber, 1997) based methods to represent such information. However, LSTM representation can hardly make correct predictions when it meets long sequences with rich linguistic information. It is necessary to find some ways that give appropriate representations to the input feature and make video understanding deeper.

In this work, some studies are taken to make video understanding deeper. Specific to this purpose, there are three points in common that are quite challenging:

- 1) Since the goal for each task is different, it is important to consider what features should be represented when facing different tasks. Then, appropriate pre-trained models shall be selected and give proper representations to the visual

and language features. Fine-tuning is also important since it fits the pre-trained model to the detailed tasks.

2) The video also has the information in temporal domains, so the appropriate feature representations in the task should combine the visual and language features in the temporal domain in order to connect them with the evolution of the video.

3) In recent years, some methods have been proposed to represent visual or language elements. Most of them are reported to have better results than previous methods. However, it is not clear how they will perform on video-related feature representation, making it a bit difficult to select the proper way to represent features and promote deeper video understanding.

Facing these challenges, it is worth taking a study of the feature representations of visual and language tasks and promote a deeper understanding of videos.

1.6 Goal, Task, and Contribution

The ultimate goal in this thesis is to make the intelligent system understand the video in a deeper manner compared with previous studies. For this goal, appropriate features related to visual and language elements are selected at first. Then, their representations are going to be improved to promote deeper video understanding in different aspects.

This study is conducted based on two different tasks for video understanding: Video question answering task and multi-modal humor prediction task. The main motivation for selecting these tasks are:

1) The videos contain both images and subtitles that are highly related to both visual and language features and are suitable for feature representations in both tasks.

2) Both tasks are relatively new research fields in artificial intelligence. Instead of processing static images, both tasks mainly process videos that contain both a set of frames and related subtitles. They not only provide another view to evaluate the performance of video understanding, but also lead the frontier of

the intelligent system.

3) Both tasks are quite challenging for the computer. To handle these tasks, the computer not only needs a joint understanding of the visual and the language features, but also has to understand how the video plot evolves in the temporal domain. Also, both tasks have relatively more input data compared with previous work. This means that the selected feature representation methods must be able to handle video very well.

In general, this thesis focuses on finding appropriate feature representations for visual and language elements towards deeper video understanding. It selects three aspects based on two different tasks and conducts the study based on various experiments. The main contributions of this thesis are summarized as follows:

First, this thesis selects different features to describe visual and language elements in two tasks respectively. For visual features, unlike previous work that mainly uses deep features to represent visual elements, this thesis selects visual concept feature for video question answering task, and selects pose and face features for humor prediction task. For language features, this thesis selects subtitles in the video that mainly reflects the dialogue between different people.

Second, this thesis finds better ways to improve the feature representations in two different tasks by considering their different characteristics. In video question answering task, BERT and other three Transformers are taken to represent the deep semantics in the visual and language inputs; in humor prediction task, BERT is used to represent subtitles while LSTM is adopted to represent pose and face features. Experiments are carried to compare the proposed methods with previous methods. Results show that the methods proposed in the thesis outperform previous methods, and promote a deeper understanding to video contents.

1.7 The Structure of the Thesis

This thesis contains 6 chapters. The main contents of each chapter are shown as follows:

Chapter 1 is the introduction part showing the background, motivation, and challenges of the research. It also introduces the main contributions and the structure of the thesis.

Chapter 2 gives brief introductions of the basic principle and the related work regarding the selected tasks, including visual question answering, video question answering, and emotion prediction.

Chapter 3 proposes the improved feature representation in video question answering task. The method first retrieves visual concept features in the form of words or phrases from video frames using faster RCNN, then uses pre-trained BERT (Devlin, Chang, Lee, & Toutanova, 2018) to represent these visual concept features along with subtitles that contain the dialogues between people. Experiments are conducted by taking different input arrangements, subtitles with/without time stamp annotations, and different maximum lengths of the input sentence into BERT models. Results show that the proposed method using BERT representation for visual and language elements finally gives a deeper understanding of videos by improving the accuracy of video question answering task.

Chapter 4 keeps looking into video question answering task. In this chapter, a deeper investigation about how Transformers work to answer questions related to video content is taken. This chapter first gives brief introductions to the specific features of different Transformers, then applies four kinds of Transformers in video question answering task. Language representations for visual concept features and subtitles are given based on different Transformers to capture semantics in both video scenes and subtitles. Experiments are performed to find the best-performed Transformer. After the experiments, further analysis is also made to show why different Transformer has different accuracy in the same dataset.

Chapter 5 introduces the dataset and the novel method to predict humor in a multimodal way. For this task, a dataset is created based on the famous sitcom *the Big Bang Theory* to predict humor, which contains video frames, subtitles, and wave envelopes from 228 episodes in 10 seasons. Unlike previous method that predicts humor based on sentences, the proposed method uses a sliding-

window-based method to predict humor segments, which are defined by the ground truth laughter in the audio track. A model is built to use pose, face, and subtitle features to predict humor. Experiments are carried in both frame level and segment level. By taking different combinations of pose and face features along with subtitle features, the best-performed combination of input features is found. Some evaluations between the proposed method and sentence-based humor prediction method are made by converting the sentence-based predictions into humor segments and comparing their performance in both frame level and segment level.

Chapter 6 is the conclusion and outlook part. This part first presents a summary of this thesis and shows the main findings. Then, it provides some suggestions regarding future studies.

Chapter 2

Related Work

Visual and language tasks have been quickly developing in these years. As described in Section 1.7, this chapter gives an introduction of the basic principle and related work regarding some related tasks.

2.1 Visual Question Answering Task

Visual question answering (VQA) is a system that answers questions about an image automatically. It usually takes an image and a related question as inputs and gives a correct answer to the question as an output (H. Xu & Saenko, 2016; Anderson et al., 2018). A basic flow chart of VQA system is shown in Fig.2.1. To answer questions correctly, VQA needs to extract the semantic information in both visual (the image) and language (the question) contents and encode them into high-dimensional vectors. Attention mechanisms are usually taken to find the correspondences between different contents, and fuse them to predict the answer (K. Xu et al., 2015). As a new field in artificial intelligence, VQA has undergone quick development in these years. Lots of related work has been proposed in different aspects to create better VQA systems. This thesis takes a look at answering question in videos, which is an improved task for visual question answering.

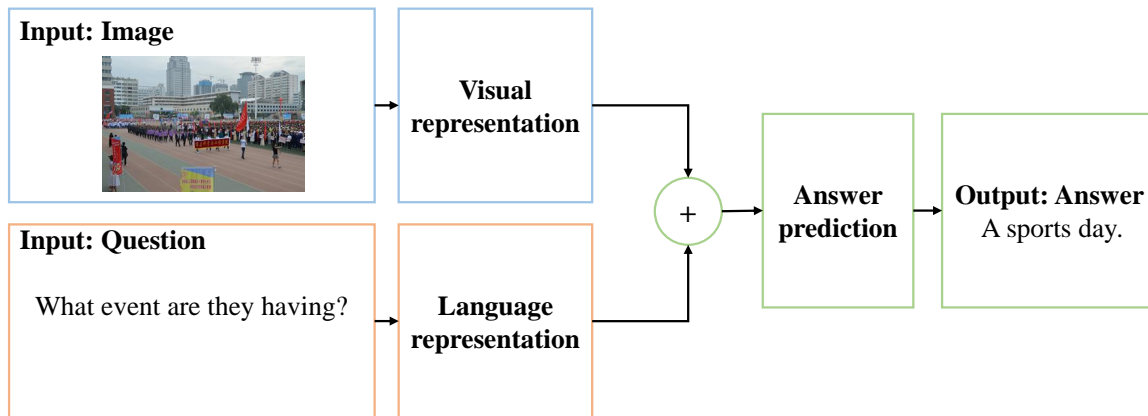


Figure 2.1. A basic flow chart of VQA system.

2.1.1 VQA Dataset

To make VQA systems better, it is necessary to have corresponding datasets for visual question answering. Lots of datasets regarding VQA have been proposed during these years, and most of them are created based on our daily life. These datasets can be generally divided into two categories by whether outside knowledge is included or not. On the other hand, some datasets for specific domains are also introduced to answer professional questions.

Datasets without Outside Knowledge Datasets without outside knowledge have questions mainly related to the objects, the actions, or the simple relationships in the image. The VQA system can answer the questions related to them by directly referring to the image. In recent years, many related datasets have been proposed. For example, Antol et al (Antol et al., 2015) propose the VQA dataset, which has an image and a natural language question related to the image; Agrawal et al (Agrawal, Batra, Parikh, & Kembhavi, 2018) present the VQA-CP dataset based on the VQA dataset (Antol et al., 2015), which allows a new setting for questions with different prior distributions of answers; Chou et al (Chou, Chao, Lai, Sun, & Yang, 2020) collect the VQA 360 dataset that contains 360-degree panorama real-world image-question-answer pairs with a variety of question types. Besides these datasets, other datasets like COCOQA

(Ren, Kiros, & Zemel, 2015), FM-IQA (H. Gao et al., 2015) and visual7W (Y. Zhu, Groth, Bernstein, & Fei-Fei, 2016) have also been introduced for the VQA task.

Datasets with Outside Knowledge Datasets with outside knowledge have questions that usually cannot be answered directly by merely using the image contents. To handle such kinds of questions, a knowledge base is usually included in the datasets to give reference to the VQA system. For example, Wang et al (P. Wang, Wu, Shen, Dick, & Van Den Hengel, 2017) propose the FVQA dataset for deeper reasoning, which contains image-question-answer triplets and questions that require external information to answer; Shah et al (Shah, Mishra, Yadati, & Talukdar, 2019) presents the KVQA dataset, whose questions require reasoning over knowledge bases to reach the answer; Marino et al (Marino, Rastegari, Farhadi, & Mottaghi, 2019) introduces the OK-VQA dataset, where the image content and external knowledge resources are required to answer the questions. Other datasets like (Basu, Shakerin, & Gupta, 2020; P. Lu et al., 2018; B. Liu et al., 2021) also provide knowledge to give assistance to a correct prediction.

Datasets for Specific Domains Some datasets are also introduced to create VQA in professional domains. These datasets usually consist of domain-specific images and questions and require the deep collaborations of people with different backgrounds to find the ground truth answer. For example, Bongini et al (Bongini, Becattini, Bagdanov, & Del Bimbo, 2020) propose a dataset for cultural heritage; Lobry et al (Lobry, Marcos, Murray, & Tuia, 2020) introduce a dataset for remote sensing and He et al (X. He, Zhang, Mou, Xing, & Xie, 2020) present a dataset for medical imaging. Other specific datasets can be found in arts (Garcia, Ye, et al., 2020), robotics (Kenfack, Siddiky, Balint-Benczedi, & Beetz, 2020), blind assistance (Gurari et al., 2018) and so on.

2.1.2 VQA Method

As a relatively new research topic, VQA has drawn much attention during these years. Many outstanding achievements have been proposed to improve the accuracy of question answering. The methods can be mainly classified into two groups by whether to improve the original feature representation or to import new information sources (like a knowledge base).

Methods to Improve the Original Feature Representations Methods in this kind focus on giving better representations of the input features in order to improve the accuracy of question answering, and usually do not import the outside information source. When the question is not explicit to the image contents, methods in this type cannot find the correct answer very well.

Some representative work has been introduced to improve the feature representations in visual question answering. For example, Xu et al (H. Xu & Saenko, 2016) present spatial memory network that stores neuron activation from different regions of the image, and uses the question to choose relevant regions for predicting the answer; Gokhale et al (Gokhale et al., 2020) introduce VQA in logic, which set up question-attention and logic-attention to understand logical connectives in the question; Xi et al (Xi et al., 2020) propose a VQA model to find the visual relationship between objects; Farazi et al (Farazi, Khan, & Barnes, 2020) use a semantic relationship parser and a Mutual and Self-attention to recognize the relationship in the image and answer the question. Besides these improvements, other strong attempts such as (Ruwa, Mao, Wang, Gou, & Dong, 2019; H. Liu et al., 2018; Chen, Li, Wang, & Long, 2018; D. Yu, Gao, & Xiong, 2018; Anderson et al., 2018), have also been proposed to find better feature representations and improve the accuracy VQA task.

Methods to Import New Information Sources Methods in this kind use some outside database to help answer the questions. In this kind of method, the system not only needs the feature representation from the image and the question, but it also has to resort to the outside database to find the related information, and com-

bine these features together to answer the question. Once the outside database covers enough information, methods in this kind will be able to answer implicit questions correctly.

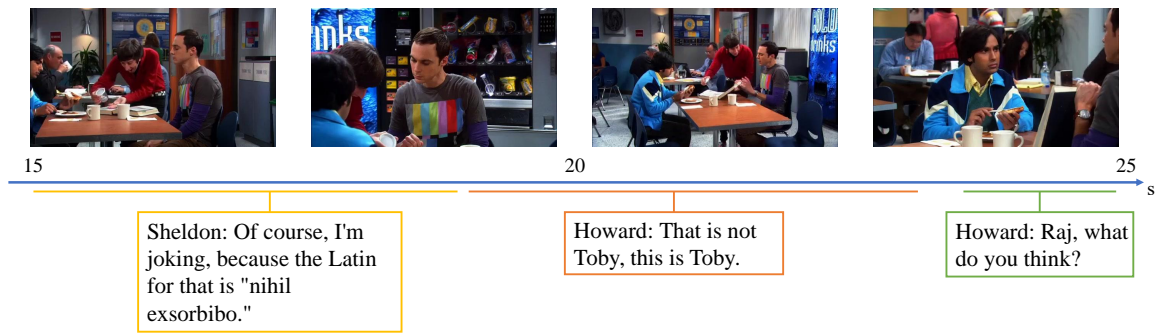
In these years, lots of work embeds knowledge base into visual question answering. For example, Narasimhan et al (Narasimhan & Schwing, 2018) propose a learning-based approach that goes straight to the facts with the help of a knowledge base; Wu et al (Q. Wu, Wang, Shen, Dick, & Van Den Hengel, 2016) gives a solution for visual question answering that combines the internal representation of the image and the information from a knowledge base; Zhu et al (Y. Zhu, Lim, & Fei-Fei, 2017) propose an approach to acquire task-driven information for visual question answering from external auxiliary data and Narasimhan et al (Narasimhan, Lazebnik, & Schwing, 2018) develop a graph convolutional network to find the correct answer by considering both features and the knowledge base. Besides them, other work (P. Wang et al., 2017; Garcia, Ye, et al., 2020; P. Wang, Wu, Shen, Hengel, & Dick, 2015) has also been proposed to build different information sources for the task.

2.2 Video Question Answering Task

Even though some visual question answering systems have a very high accuracy in answering questions about the static image, when they face videos, their performances still drop a lot. Hence, this thesis takes research regarding methods that can answer questions in videos.

To correctly answer questions in videos, video question answering (video-QA) task (shown in Fig. 2.2) has been studied. Different from question answering based on static images, video-QA task needs to understand the video and answer questions based on its contents. This requirement gives some unique challenges towards video-QA tasks:

First, a video consists of a set of frames instead of a static image. They connect with each other in temporal domains and contain abundant information. The video-QA system needs to understand the temporal evolution between different video frames in order to make a correct prediction (L. Zhu, Xu, Yang, &



Q: Why is Howard frustrated when he is talking to Sheldon?

Figure 2.2. An example of question answering in video

Hauptmann, 2017; Mun, Hongsuck Seo, Jung, & Han, 2017).

Second, a video also has subtitles, which contain the transcription of what the characters are saying. It is also important for the video-QA system to understand them because they often convey complementary information that is not shown in video frames (K.-M. Kim, Heo, Choi, & Zhang, 2017; Garcia, Otani, Chu, & Nakashima, 2020; Lei et al., 2018).

Based on these challenges, a video-QA system needs a joint understanding of the video frames, the associated subtitles along with the question and candidate answers to make a correct prediction. Thus, it processes relatively more input data than the standard VQA system. Specific methods are needed to extract and represent the visual and language contents in a large amount.

2.2.1 Video-QA Datasets

There are many datasets related to video-QA task. Tapaswi et al (Tapaswi et al., 2016a) introduce the MovieQA dataset based on movies to evaluate automatic story comprehension from both video and text information; Kim et al (K.-M. Kim et al., 2017) create the Pororo dataset based on children's cartoon films called *Pororo*; Lei et al build the TVQA (Lei et al., 2018) dataset and the TVQA+ (Lei et al., 2019) dataset based on movies to examine the abilities to localize relevant moments in video-QA systems; Colas et al (Colas et al., 2019) propose a dataset called TutorialVQA based on instructional videos to

check the ability to identify a span of a video segment as an answer. Some other datasets, like videoQA (Zeng et al., 2017), SVQA (Song, Shi, Chen, & Han, 2018), knowitVQA (Garcia, Otani, et al., 2020) and iVQA (A. Yang, Miech, Sivic, Laptev, & Schmid, 2020) have also been introduced to answer questions in videos. This thesis uses the datasets provided in (Lei et al., 2018) and (K.-M. Kim et al., 2017).

2.2.2 Video-QA Methods

In the last few years, some work related to the improvement of video-QA has also been proposed. The methods can be mainly classified into two categories by the input modality they studied. This thesis contributes to the improvement of video question answering task by giving better representations to the visual elements.

Improvements on Visual Representations Improvements on visual representations try to find better ways to represent visual features or the evolution between video frames in video-QA tasks. For example, Huang et al (D. Huang et al., 2020) represent the location information of objects in the video as location-aware graphs and let the system better understand object interaction; Ye et al (Ye et al., 2017) propose the attribute-augmented attention learning framework that enables the joint frame-level attribute detection and unified video representation for video question answering; Zhu et al (L. Zhu et al., 2017) present an encoder-decoder approach with recurrent neural networks to learn the temporal structures of videos. Some other explorations of visual representations can be found in (Zeng et al., 2017; Zhao, Yang, Cai, He, & Zhuang, 2017; J. Gao, Ge, Chen, & Nevatia, 2018).

Improvements on Language Representations Besides improvements on visual representations, some methods are also proposed to find better language representations for video-QA task. For example, Lei et al (Lei et al., 2019) improve the language representation by using BERT to represent language features in video-QA; Yu et al (Y. Yu, Kim, & Kim, 2018) present an approach to measure seman-

tic similarity between video sequence and language sentence to learn the effective multimodal matching between sequence data; Noa et al (Garcia, Otani, et al., 2020) propose a novel video understanding task by fusing knowledge-base and video question answering to let the system understand knowledge-based questions. Other related methods can be found in (J. Kim, Ma, Kim, Kim, & Yoo, 2019; Mun, Cho, & Han, 2020; Lei et al., 2018), which also improve the language representations of video-QA task.

2.3 Emotion Prediction Task

Emotions show the status of pleasure or displeasure, and can be expressed in the ways like facial expressions, gestures, speech and text. Emotions are often intertwined with personality, disposition, and creativity (Averill, 1999), giving affections to how people treating others. This thesis is going to handle humor, one of the common emotions made by human, to give some inspiration to future work.

It is an important task to use artificial intelligence to understand people's emotions because the artificial intelligence system is designed to meet various needs of people. By understanding people's emotions, an artificial system knows how people feel under different situations, and will ultimately take emotions into their decision strategies and respond to people's needs in different ways.

However, emotion prediction is a challenging task in both computer vision and natural language processing communities. Since emotions are expressed in various ways, a computer needs a deep understanding of not only signals in each modality but also their combinations to predict how they affect people's emotions. Recently, some research efforts have been made to give predictions to people's emotions in both single modality and in multiple modalities. A typical concept of emotion prediction task in multiple modalities is shown in Fig. 2.3.

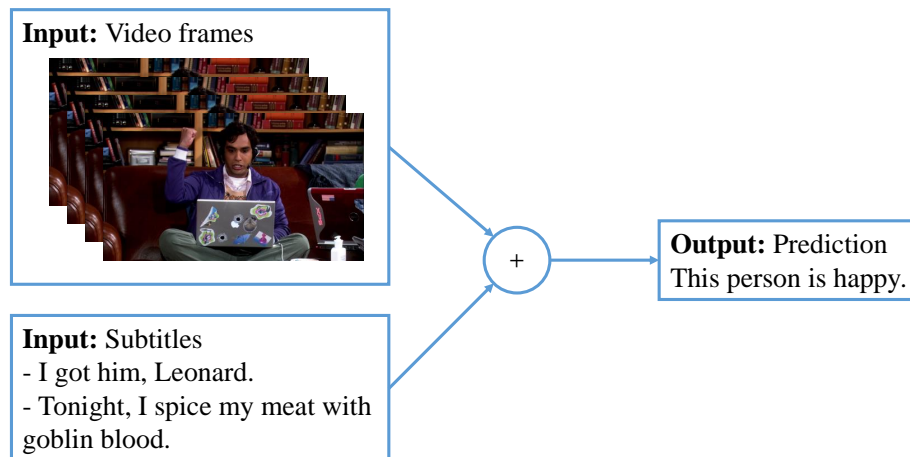


Figure 2.3. A typical concept of emotion prediction task in multiple modalities

2.3.1 Emotion Prediction Datasets

Some datasets have been introduced in these years to predict emotions. These datasets can be mainly divided into single modality datasets and multiple modalities datasets.

Single Modality Datasets Single modality datasets are mainly used to predict emotions in the linguistic modality or the visual modality. Some representative single modality datasets have been proposed in these years. For example: Czapla et al (Czapla & Howard, 2019) present a dataset based on the Spanish tweets on Twitter to predict humor; Pestian et al (Pestian et al., 2012) introduce a dataset based on clinical text and annotated notes to classify the emotions of those who have died by suicide; Lucey et al (Lucey et al., 2010) present a dataset called extended cohn-kanade that uses facial action unit to predict people's emotions. Some other datasets can be found in (Hossain, Krumm, & Gamon, 2019; Engelthaler & Hills, 2018; Kleinberg, van der Vegt, & Mozes, 2020; You, Luo, Jin, & Yang, 2016).

Multiple Modalities Datasets Multiple modalities datasets contain information from other modalities, like video and audio information. Some multiple modalities datasets have been proposed in these years. Hasan et al. (Hasan et al., 2019)

propose the UR-Funny dataset, which focuses on acoustic, visual, and linguistic information in ted talks to predict humor; Jiang et al (Jiang, Xu, & Xue, 2014) introduce a dataset collected from online videos to predict emotions, which includes the input in the visual and acoustic modality; Shen et al (Shen, Wang, Duan, Li, & Zhu, 2020) propose a dataset that uses acoustic, visual, and linguistic information in movies to give reasonings to emotions. Some other datasets have also been proposed by taking the visual, language, and audio (Mittal et al., 2020; C. Liu, Tang, Lv, & Wang, 2018; Kayatani et al., 2021; Patro et al., 2021) into consideration. In this thesis, the proposed dataset is related to humor prediction in a multi-modal way.

2.3.2 Emotion Prediction Methods

Single Modality Emotion Prediction Single modality emotion prediction methods mainly predict the emotion in the linguistic or the visual modality. For example: Fan et al. (Fan et al., 2020) uses an internal and external attention neural network for a humor prediction task in short texts; Ruwa et al (Ruwa et al., 2019) introduce a novel neural network model that combines the image features, the question, and the mood to produce a mood-based answer; Tan et al (Tan et al., 2017) present an approach for group-level emotion recognition based on two types of Convolutional Neural Networks in the image; Czapla et al (Czapla & Howard, 2019) apply a pre-trained language model to predict humor in Spanish tweets.

Multiple Modalities Emotion Prediction Multiple modalities emotion prediction methods give their predictions by combining information from different modalities together. Some multiple modalities methods have been proposed to predict people's emotions in recent years. For example, Hasan et al. (Hasan et al., 2019) attempt to understand humor based on the subtitle, visual, and audio features in TED talk videos; Huang et al (J. Huang et al., 2017) apply features in visual, acoustic, and linguistic modality and conduct fusion across them to know the emotional state of the speaker; Shen et al (Shen et al., 2020) propose an attention-based reasoning approach to model the intra-personal emotion con-

texts, inter-personal emotion propagation, and the personalities of different individuals. This thesis also introduces a method that uses inputs from multiple modalities to predict humor.

2.4 Feature Representation Methods

Feature representation not only builds the bridge between humans and computers, but also connects the original inputs and the prediction together. To promote a deeper understanding of videos, feature representation needs to convert the visual and language elements into the format that can be handled by computers. In recent years, feature representation has become a popular research topic. Many methods have been introduced based on visual or language inputs to give better representations to the input features.

Visual Feature Representation Methods Visual feature representation methods transform the visual features into vectors that can be handled by computers. In recent years, many methods have been proposed to explore better visual features. For example, in deep residual network (K. He et al., 2016a), a deeper residual learning framework is introduced to ease the training of networks; in bottom-up and top-down attention (Anderson et al., 2018), attention weights are calculated at the level of objects and other salient image regions; in multi-level scene description network (Y. Li, Ouyang, Zhou, Wang, & Wang, 2017), object, phrase, and caption regions interact with each other via a dynamic graph based on their spatial and semantic connections; in visual semantic parsing (Zareian, Karaman, & Chang, 2020), a generalized formulation of scene graph is proposed to disentangle entity and predicate recognition. Some other work (K. He, Zhang, Ren, & Sun, 2016b; Goyal, Khot, Summers-Stay, Batra, & Parikh, 2017; J. Lu, Xiong, Parikh, & Socher, 2017) has also been proposed to give representations to the visual features.

Language Feature Representation Methods Language feature representation methods associate each word in a sentence with a real-valued vector, and make the pro-

cessing of natural language easier. Methods of this kind have been widely used in various tasks. Different methods have been proposed in the last few years. For example, GloVe (Pennington, Socher, & Manning, 2014) is presented to leverage statistical information at the word level by training only on non-zero elements in a word-word co-occurrence matrix. Skip-thoughts vectors (Kiros et al., 2015) generate language representations at sentence level, providing a generic, distributed encoder to reconstruct the surrounding sentences of an encoded passage. Besides these methods, many Transformers have been proposed for sentence feature representation. For example, BERT (Devlin et al., 2018) is proposed to give deep bidirectional representations of unlabeled text; XLNet (Z. Yang et al., 2019) enables bidirectional context learning by maximizing the expected likelihood over all permutations of the factorization order; RoBERTa (Y. Liu et al., 2019) is an improved recipe for training BERT models; and ALBERT (Lan et al., 2019) lowers the memory consumption and increases the training speed of BERT. Some other language representations (McCann, Bradbury, Xiong, & Socher, 2017; Conneau, Kiela, Schwenk, Barrault, & Bordes, 2017; Turian, Ratinov, & Bengio, 2010; J. Lu, Batra, Parikh, & Lee, 2019) have also been proposed, aiming at representing language features in words or sentences better.

Chapter 3

BERT Representations for Video Question Answering

In accordance with the outline presented in Section 1.7, this chapter provides a detailed introduction of the proposed method of using BERT to improve the feature representations and answer questions related to video contents. The proposed method uses pre-trained BERT to represent visual semantics obtained from video frames. Different experiments are also conducted to evaluate the performance of the proposed method in video question answering task.

3.1 Introduction

Video question answering is a task that answers questions about a video. It usually takes the video frames, the corresponding subtitles, and the question as inputs, and predicts a correct answer to the question as an output (Lei et al., 2018, 2019; K.-M. Kim, Choi, Kim, & Zhang, 2018). Video question answering has drawn much attention during these years, some strong attempts to the improvement of this task have been proposed (Lei et al., 2018; Tapaswi et al., 2016a; K.-M. Kim et al., 2017; Y.-C. Wu & Yang, 2008; Y.-C. Wu, Chang, & Lee, 2004; Tapaswi et al., 2016b; Zhao et al., 2018).

In order to make the prediction correctly, the computer needs to have a deeper understanding of the video contents. So far, most models that have been proposed to answer questions in videos (Lei et al., 2018; Tapaswi et al., 2016a; K.-M. Kim et al., 2017) will encode the language information from the questions

and the subtitles using RNNs, especially long short-term memory (LSTM) networks (Hochreiter & Schmidhuber, 1997). LSTM networks encode each word sequentially and form a representation for the whole sentence. Then, this representation will be conveyed to a classifier to produce the output for a correct answer. However, LSTM representations might fail to predict answers when the semantic relationships are contained in long text sequences, such as the ones that appear in the subtitles of long video clips (e.g. about 30 seconds).

On the other hand, BERT, a powerful network based on language Transformers (Devlin et al., 2018), has been proposed for handling long sequences. BERT has been shown to outperform LSTMs in several natural language processing tasks (Devlin et al., 2018; Gong et al., 2019; Z. Li, Ding, & Liu, 2019). However, studies of feature representation with BERT in video question answering task are very limited. Lei et al (Lei et al., 2019) proposed a method that uses off-the-shelf BERT to extract pre-trained representations from the language information (i.e. subtitles, questions, and answers), but still used LSTM for visual feature representations. With the goal of finding a way to better represent visual features, this chapter goes one step further and also represents visual features with BERT.

In this chapter, a study about BERT representations for deep video understanding is performed. The study selects video question answering task to reflect how well the proposed model can understand videos. The proposed method uses BERT (Devlin et al., 2018) to represent both visual and language features in video-QA task by capturing semantic information from both visual and language flows in the video clips. This method goes one step further than previous work (Lei et al., 2019) in that it not only fine-tunes the network for the task of interest, but also relies on BERT to encode both the language and the visual information, which is shown in Fig. 3.1.

The main contributions of the proposed work are:

- 1) BERT (Devlin et al., 2018) is used to represent both visual and language features in video-QA task.
- 2) BERT is adopted to encode not only the information from subtitles, but also the visual information from the video frames.

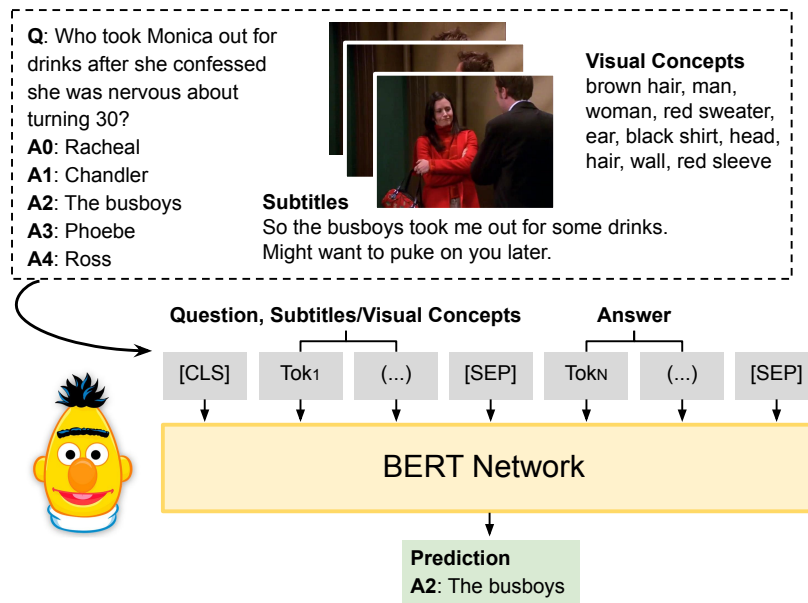


Figure 3.1. The proposed model for video-QA based on BERT.

3) Several experiments are conducted to analyze the accuracy of BERT representations in video question answering.

3.2 Prior Work

Some prior work is shown mainly in two aspects: video question answering and language representations.

3.2.1 Video Question Answering

Video question answering system is introduced to answer questions related to the video. To correctly answer the question, a joint understanding of the question, the video frames, and the associated subtitles is required. Some work related to video question answering has been proposed in recent years, for example: Lei et al (Lei et al., 2018) introduce a method to embed the visual and language features with LSTM and answer video-based questions, whereas the same authors improve the results in (Xi et al., 2020) with the application of BERT in representing subtitles features, while they keep using LSTM for visual features; Garcia et al (Garcia, Otani, et al., 2020) present a video understanding

task by fusing external knowledge and video-QA together. They use BERT to represent question, candidate answer and knowledge, and use bag-of-concept to represent visual features; Kim et al (J. Kim et al., 2019) propose a video question answering framework that requires a simultaneous retrieval of the relevant moments and referenced visual concepts. Their features are mainly represented with LSTM; Zhu et al (H. Xu & Saenko, 2016) study video-QA and carry inferences and predictions in the temporal domain based on representing video features with gated recurrent unit. Other related work can be found in (Y.-C. Wu & Yang, 2008; Y.-C. Wu et al., 2004; Tapaswi et al., 2016b; Zhao et al., 2018), which also improves the system in different aspects. The proposed work distinguishes from previous studies in that it is using BERT to model the information captured from the video clips.

3.2.2 Language Representation

Language representations are widely used for processing the language information in question answering models. Generally, models that represent language features associate each word in a sentence with a real-valued vector. Different methods for language representations have been proposed in the last few years. For example: Pennington et al (Pennington et al., 2014) propose GloVe to leverage statistical information by training only on the non-zero elements in a word-word co-occurrence matrix; Socher et al (Socher et al., 2013) propose sentiment tree banks along with recursive neural tensor network to represent the meaning of longer phrases in a principled way; McCann et al (McCann et al., 2017) propose an LSTM encoder from attentional sequence-to-sequence model to contextualize word vectors, and show that these context vectors improves the performance in a wide range of related tasks; Kiros et al (Kiros et al., 2015) propose Skip-Thoughts to let sentences with semantic and syntactic properties being mapped to similar vector representations, and hence, learn a generic, distributed sentence encoder in an unsupervised way. Besides these methods, some other language representations (Conneau et al., 2017; Turian et al., 2010) have also been proposed.

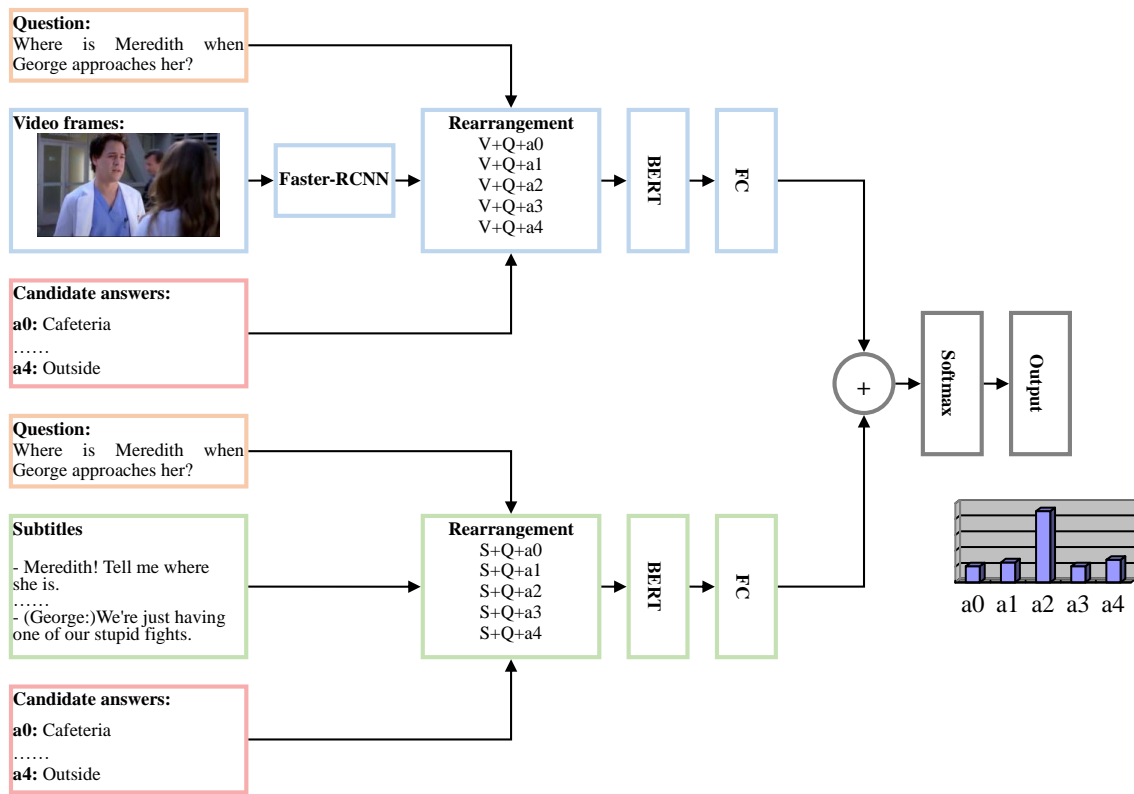


Figure 3.2. The framework of the proposed method.

3.3 The Proposed Method

The flow chart of the proposed model is shown in Fig.3.2, where video-QA is addressed as a multiple-choice task by following previous work (Lei et al., 2018; K.-M. Kim et al., 2017). The proposed method builds a classifier to find the correct answer to each question based on the input features. In this method, the visual semantic information is first extracted from each video frame as visual concepts using Faster-RCNN (Anderson et al., 2018) fine-tuned on the Visual Genome dataset (Krishna et al., 2017). Then, the extracted visual concepts and the subtitles are processed in two independent flows along with the question and candidate answers. In each flow, a fine-tuned BERT network is applied independently, which is used to predict the correct answer according to its inputs. The outputs of the two flows are jointly processed to obtain the final answer prediction.

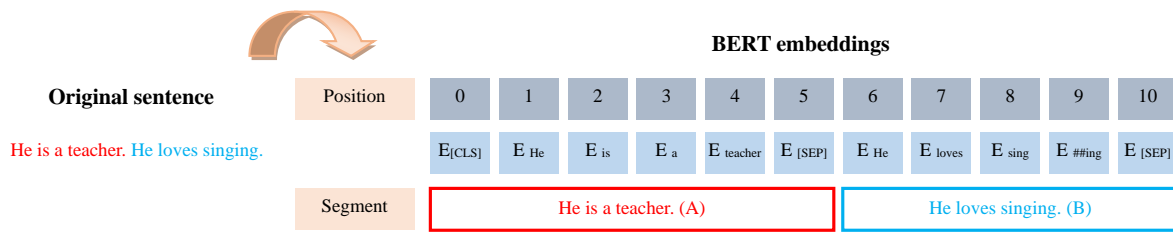


Figure 3.3. An example of input representation using BERT.

3.3.1 Introduction to BERT

BERT is the short for bidirectional encoder representation from transformers. It is a language representation model designed to extract pre-trained deep bidirectional representations from the unlabeled text by jointly conditioning on both left and right context in all layers (Devlin et al., 2018). Thus, pre-trained BERT model can be fine-tuned with just one additional output layer, and create state-of-the-art models for a wide range of tasks, such as question answering and language inference.

The BERT representation for a given sentence contains three embeddings, namely token embeddings, segment embeddings, and position embeddings. Token embeddings show the embedding of each word in the sentence, segment embeddings denote the belonging of each token (A: former sentence; B: latter sentence), and position embeddings give the position order for each word in the sentence. The first token in front of every former sentence is [CLS], which is used to obtain the output in classification tasks. Between the former sentence and the latter sentence, a [SEP] token is added to indicate the separation. An example of BERT representation for sentence *He is a teacher. He loves singing* is shown in Fig 3.3.

To obtain a deep bidirectional representation, BERT uses masked language modelling and a binarized next sentence prediction during its pre-training. Masked language modelling will mask some of the input tokens at random, and then gives predictions to those masked tokens. To understand the relations between two sentences, a binarized next sentence prediction is also taken in the pre-training to give predictions of whether a certain sentence is the next sentence of

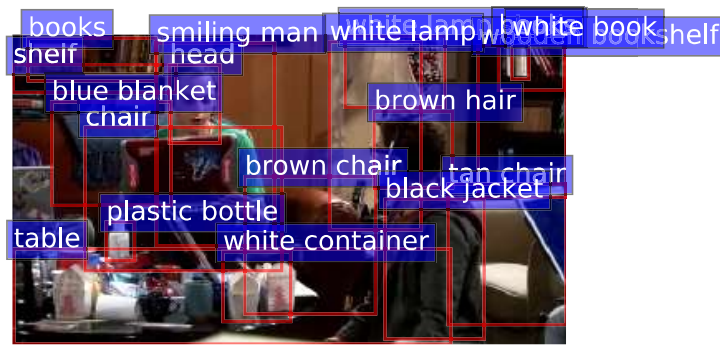


Figure 3.4. Detected visual concept features using Faster-RCNN (Anderson et al., 2018) pre-trained on the Visual Genome dataset.

another sentence. The pre-training corpora are the BooksCorpus (800M words) (Y. Zhu et al., 2015) and English Wikipedia (2,500M words).

3.3.2 Feature Representations

The proposed method retrieves information from visual and language features respectively and uses two independent BERT networks to predict the correct answer to each question related to videos.

Visual Representations Recent work (Lei et al., 2018; Yin & Ordonez, 2017) has found that using detected object labels as input has comparable or better performance than using CNN features directly in image captioning and video-QA tasks. Motivated by them, the proposed method uses detected object labels to represent features from images, which are named visual concept features. These visual concept features are obtained by setting up some bounding box object detectors in images. They contain both objects and attributes, such as *grey pants*, *woman*, *blonde hair*, etc. An example of visual concept features is shown in Fig. 3.4.

The visual concept features are extracted from each video frame using Faster R-CNN (Anderson et al., 2018) fine-tuned on the Visual Genome (Krishna et al., 2017) dataset as in (Lei et al., 2018). The video frames are extracted from original videos with a sampling rate of 3 fps. In every extracted frame, the vi-

sual concept features are represented by corresponding words or phrases. The visual concept features from all the frames are concatenated together. Duplicated features are then removed to obtain the unique visual concept features from a whole scene v . Then, the unique visual concept features v are concatenated with the question q and each candidate answer a_i ($i=0,1,2,3,4$) to form a single rearranged string c_i . Each rearranged string is tokenized to obtain the sequence T_{c_i} .

$$c_i = [v, q, a_i] \quad (3.1)$$

$$T_{c_i} = \text{tokenize}(c_i) \quad (3.2)$$

Here, the concatenation of v and q , $[v, q]$, is set as the former sentence of BERT embedding and $[a_i]$ as the latter sentence of BERT embedding. As BERT can only process no more than L tokens at a time, the last token(s) in the former and latter sentences are truncated until the number of words in T_{c_i} is no more than the maximum number of words L .

Next, T_{c_i} is conveyed into the BERT network. The output is a matrix V_{c_i} that represents each word in the input sequence as vector. The output vector corresponding to the [CLS] token $V_{c_i}^0$ is conveyed into a fully connected layer to obtain the visual flow prediction R_{c_i} for answer i , where F_c is a trainable parameter.

$$V_{c_i} = \text{BERT}_c(T_{c_i}) \quad (3.3)$$

$$R_{c_i} = F_c V_{c_i}^0 \quad (3.4)$$

Language Representations The subtitles contain what the characters are saying, and are an important information source for question answering. Similar to the visual flow, the subtitles s , the question q and the candidate answer items a_i ($i=0,1,2,3,4$) are concatenated to form the rearranged string w_i in the language flow. Then w_i is tokenized to form the sequence of tokens T_{w_i} , which is conveyed

into the BERT network to obtain V_{w_i} , a matrix that represents each word in the input sequence as vector. The language flow prediction R_{w_i} for answer i is obtained as:

$$w_i = [s, q, a_i] \quad (3.5)$$

$$T_{w_i} = \text{tokenize}(w_i) \quad (3.6)$$

$$V_{w_i} = \text{BERT}_w(T_{w_i}) \quad (3.7)$$

$$R_{w_i} = F_w V_{w_i}^0 \quad (3.8)$$

3.3.3 Answer Prediction

To obtain the final answer prediction, the predictions of visual and language flows for each candidate answer are summed together to obtain R_{p_i} . Softmax is adopted to convert the summed vector into the answer scores R_f :

$$R_{p_i} = R_{c_i} + R_{w_i} \quad (3.9)$$

$$R_p = [R_{p_0}, R_{p_1}, R_{p_2}, R_{p_3}, R_{p_4}] \quad (3.10)$$

$$R_f = \text{softmax}(R_p) \quad (3.11)$$

Finally, the answer index with the maximum score is selected as the final predicted answer a_p , with:

$$p = \text{argmax}(R_f) \quad (3.12)$$

Table 3.1

The training parameters in the experiment.

Parameters	Setting
Learning Rate	2×10^{-5}
Number of Epoch	10
Training Batch Size	8
Inferring Batch Size	8
Max Number of Tokens (L)	128
Optimizer	Adam
Weight Decay	1×10^{-5}

3.4 Experimental Results

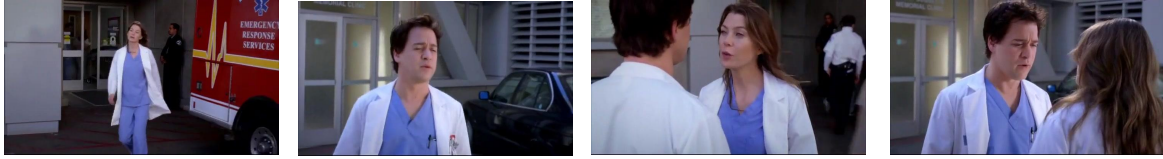
3.4.1 Experimental Settings

The proposed method is implemented based on Python 3 and Pytorch. The training parameters in the experiments are shown in Table 3.1. This method uses `bert-base-uncased` model with 12 layers, 768 hidden sizes, 12 self-attentions and 110 million parameters. It makes no distinction between upper case and lower case tokens. The accuracy of question answering is adopted as the evaluation metric of each method.

3.4.2 Datasets

Two video-QA datasets are used in the experiments: TVQA (Lei et al., 2018) and Pororo (K.-M. Kim et al., 2017). The TVQA dataset is created based on six TV shows with 152,500 question-answer pairs (Q/A pairs) from 21,800 clips, while the Pororo dataset is created based on a children’s cartoon video series called *Pororo* with 8,834 Q/A pairs from 171 episodes. Both datasets contain three parts: The video frame, the subtitle, and question/answers. The video frame is resampled from the original video, the subtitle is provided corresponding to each video scene and the questions are formed as multiple-choice style to let the computer select one correct answer out of five candidate answers. Examples of the TVQA and the Pororo datasets are shown in Fig. 3.5 and Fig. 3.6 respectively. The computer need to have a joint understanding of visual and language features in order to give correct answers to the questions in both

Video:



Subtitle:

00:01:16,735 --> 00:01:19,033
Meredith! Tell me where she is.
00:01:19,104 --> 00:01:21,800
- I can't! If she wanted you to know...
- She does. She...
00:01:22,340 --> 00:01:25,605
- George: We're just having one of our stupid fights.

Q: Where is Meredith when George approaches her?

a0: Cafeteria
a1: Hallway
a2: Car
a3: Patients room
a4: Outside (✓)

Figure 3.5. Example of the TVQA dataset.

Video:



Subtitle:

In a cold country in a small village in the forest lives a curious little penguin named Pororo.
Hi I am Pororo.
.....
Today, the little penguin Pororo made friend with a baby dinosaur Crong.

Q: Who lived in a cold country in a small village in the forest?

a0: Loopy lived in the village.
a1: Pororo lived in the village. (✓)
a2: Poby lived in the village.
a3: Crong lived in the village.
a4: Eddy lived in the village.

Figure 3.6. Example of the Pororo dataset.

datasets.

In the TVQA dataset, the long episodes are divided into short segments whose lengths are 60 sec to 90 sec, the corresponding video and language elements are annotated with time stamps in each Q/A pair to denote the related segment of the question. There is a `<eos>` mark at the end of each sentence in the subtitle implying the sentence is over. The original test set in the TVQA dataset has no labels to the correct answer, and one needs to submit his results to the test server for its test accuracy. Since the number of total submissions to the TVQA test server is limited, the experiment splits 15,253 Q/A pairs from the training set to form a test* set, while the validation set is kept the same. For

a fair comparison, some results on the official test set are also reported. In the Pororo dataset, descriptions of the video scenes are also provided. To compare whether the proposed method is better than previous-proposed methods, results on TVQA (Lei et al., 2018), STAGE (Lei et al., 2019) and MDAM (K.-M. Kim et al., 2018) are reported. It is also interesting to compare the results of the proposed method using the MovieQA dataset (Tapaswi et al., 2016b), and refer to several methods like MovieQA baseline (Tapaswi et al., 2016b) and PAMN (J. Kim et al., 2019), but this dataset was not available when this research is carried.

3.4.3 Input Sequences

The input sequences with visual concepts, subtitles, questions and candidate answer are rearranged in three ways. The difference between each arrangements only lies in the separation characters between visual concepts/subtitles and questions:

$$\begin{aligned}
 &1) [\text{CLS}] + \text{V/S} + \text{Q} + [\text{SEP}] + \text{A} \\
 &2) [\text{CLS}] + \text{V/S} + . + \text{Q} + [\text{SEP}] + \text{A} \\
 &3) [\text{CLS}] + \text{V/S} + [\text{SEP}] + \text{Q} + [\text{SEP}] + \text{A}
 \end{aligned} \tag{3.13}$$

Where V represents visual concepts, which corresponds to v in Section 3.3, S represents subtitles, which corresponds to s , Q represents question, which corresponds to q , and A represents answer, which combines all a_i . V/S indicates that both visual concepts and subtitles are taken in visual and language flows. Ablation studies are conducted by removing either visual concepts (S+Q+A) or subtitles (V+Q+A).

3.4.4 Results on the TVQA Dataset

Results on the TVQA dataset are shown in Table 3.2. The TVQA dataset provides two modes for evaluation: One is to use the provided time stamp annotations, and another is to use full-length features. In time stamp annotation mode,

Table 3.2

Accuracy (in %) of the proposed method on TVQA dataset with time stamp annotations. Note that one only has limited chances to submit their results to the test server for evaluation, thus only some representative results of the proposed method are shown to public.

Input	Name	Visual	Language	Val	Test*	Test
Q+A	TVQA (Lei et al., 2018)	-	GloVe + LSTM	42.77	-	43.50
	Proposed [CLS]+Q+[SEP]+A	-	BERT	46.88	47.54	-
V+Q+A	TVQA (Lei et al., 2018)	GloVe + LSTM	-	45.03	-	45.44
	Proposed [CLS]+V+Q+[SEP]+A	BERT	-	48.91	49.45	-
	Proposed [CLS]+V+.+Q+[SEP]+A	BERT	-	48.95	49.23	-
	Proposed [CLS]+V+[SEP]+Q+[SEP]+A	BERT	-	48.74	49.53	-
S+Q+A	TVQA (Lei et al., 2018)	-	GloVe + LSTM	65.15	-	66.36
	Proposed [CLS]+S+Q+[SEP]+A	-	BERT	70.08	69.42	-
	Proposed [CLS]+S+.+Q+[SEP]+A	-	BERT	70.09	70.13	-
	Proposed [CLS]+S+[SEP]+Q+[SEP]+A	-	BERT	70.65	70.22	-
V/S+Q+A	TVQA (Lei et al., 2018)	GloVe + LSTM	GloVe + LSTM	67.70	-	68.48
	STAGE (Lei et al., 2019)	GloVe + LSTM	BERT	70.50	-	70.23
	Proposed [CLS]+V/S+Q+[SEP]+A	BERT	BERT	72.06	72.54	73.57
	Proposed [CLS]+V/S+.+Q+[SEP]+A	BERT	BERT	72.41	72.23	72.71
	Proposed [CLS]+V/S+[SEP]+Q+[SEP]+A	BERT	BERT	72.35	72.50	73.06

only the visual concepts and subtitles corresponding to time stamp annotation of each question are taken as the input. In full-length mode, all the visual concepts and subtitles in the videos are taken as the input. In the experiments, the input elements under provided time stamp annotations are used. To compare the results with previous methods, some results using other methods are also reported, including results of the TVQA model (Lei et al., 2018), which uses LSTM to represent both visual and language features, and STAGE (Lei et al., 2019), which uses LSTM to represent visual features, and BERT to represent language features. From the results, when BERT is used to represent both visual and language features, the proposed method obtains an accuracy up to 5.09% higher than the one obtained with TVQA, and up to 3.34% higher than STAGE.

3.4.5 Results on the Pororo Dataset

The results on the Pororo dataset are shown in Table 3.3. For a fair comparison, those video scene descriptions are ignored, while only the video frames and subtitles are taken into consideration. The proposed method is compared with MDAM (K.-M. Kim et al., 2018) model. The dataset is also used under TVQA

Table 3.3

Accuracy (in %) of the proposed method on Pororo dataset.

Input	Name	Model	Val	Test
S+Q+A	MDAM (K.-M. Kim et al., 2018)	LSTM	-	42.50
	TVQA (Lei et al., 2018)	LSTM	37.60	33.90
	Proposed [CLS]+S+Q+A	BERT	55.57	52.54
	Proposed [CLS]+S+.+Q+A	BERT	48.93	50.04
	Proposed [CLS]+S+[SEP]+Q+A	BERT	56.49	55.41
V/S+Q+A	MDAM (K.-M. Kim et al., 2018)	LSTM	-	48.90
	TVQA (Lei et al., 2018)	LSTM	37.78	42.53
	Proposed [CLS]+V/S+Q+A	BERT	48.93	48.42
	Proposed [CLS]+V/S+.+Q+A	BERT	54.14	53.79
	Proposed [CLS]+V/S+[SEP]+Q+A	BERT	52.45	52.18

(Lei et al., 2018) framework to know how TVQA method performs. Finally, the proposed method obtains an accuracy up to 4.89% higher than MDAM and up to 11.26% higher than TVQA. Note that the experiment does not report the results on DEMN model (K.-M. Kim et al., 2017) as they use the video scene descriptions and will lead to unfair comparison.

3.4.6 Ablation Study

From Table 3.2, when visual concepts along with questions and answers are used on the TVQA dataset, the accuracy improves by about 4% compared to those using LSTM. It can also be found that the use of subtitles makes a big leap in the accuracy of video-QA tasks. When subtitles, questions, and answers are included in the input on the TVQA dataset, the accuracy increase by more than 20% compared to those using questions and answers only. These results indicate that a strong visual and language representation model helps understand the video better.

From Table 3.3, BERT improves the accuracy by about 20% under TVQA framework when only subtitles, questions, and answers are taken as input. When visual concept features are used along with subtitles, the accuracy slightly drops. The reason might be that Pororo is a cartoon-based dataset, while the Faster RCNN in the proposed method is pre-trained on visual genome, which is a real-world-based dataset. Hence, the features are not so accurate in cartoon-based

Table 3.4

Training time of one epoch in TVQA dataset (h:mm)

Input	Method	Training time
V+Q+A	LSTM	2:15
	BERT	2:57
S+Q+A	LSTM	2:17
	BERT	2:58
V/S+Q+A	LSTM	4:15
	BERT	5:28

films. But generally, the proposed method still outperforms the previous methods and understands the video better.

3.5 Discussion

3.5.1 Training Time

It is important to know how long does it take for the proposed method to train one epoch of input data. The experiments are conducted on a desktop computer with Intel Core i7 8700K CPU (3.70GHz), 32G RAM and Nvidia TITAN RTX GPU. The training time for one epoch with different inputs using the TVQA dataset is reported and compared with the previous method using LSTM (Lei et al., 2018) in Table 3.4.

From the table, it can be seen that the training time of the method based on BERT is longer than the method based on LSTM. This is because BERT encodes the answers along with the question and visual concepts or subtitles, respectively, making the tokenization and training both time-consuming and memory-consuming. In the real experiments, the best validation accuracy in both methods is usually obtained within 2 epochs in both flows.

3.5.2 Evaluation with Full-Length Elements

It is also interesting to know how well the proposed method performs when full-length elements are taken as input. Hence, full-length elements (i.e. visual concepts and subtitles without time stamp annotations) instead of the time stamp

Table 3.5

Accuracy results (in %) of the proposed method on the TVQA dataset using full-length subtitles (without timestamps annotations). Note that one only has limited chances to submit their results to the test server for evaluation, thus only some representative results of the proposed method are shown to public.

Input	Visual and Language Representation	Val	Test*	Test
	TVQA (Lei et al., 2018)	64.42	-	66.46
	PAMN (J. Kim et al., 2019)	-	-	66.77
	STAGE (Lei et al., 2019)	68.56	-	69.67
V+S+Q+A	Proposed [CLS]+V/S+Q+[SEP]+A	60.17	60.48	-
	Proposed [CLS]+V/S+.+Q+[SEP]+A	60.42	60.47	-
	Proposed [CLS]+V/S+[SEP]+Q+[SEP]+A	61.97	61.67	-
	Proposed [CLS]+V/S+Q+[SEP]+A with pruning	63.07	62.77	62.72
	Proposed [CLS]+V/S+.+Q+[SEP]+A with pruning	62.05	61.94	-
	Proposed [CLS]+V/S+[SEP]+Q+[SEP]+A with pruning	62.87	62.54	-

annotated elements are taken as the input of the proposed model. In order to convey the tokens as much as possible to the neural network, the maximum number of tokens per input, L , is set as 512. The results of question answering are shown in Table 3.5.

From the results, it can be found that the best validation accuracy of the proposed method is only 61.97%, which is 2.45% lower than the TVQA model without time stamp annotations (Lei et al., 2018), and is 6.59% lower than STAGE (Lei et al., 2019). The drop in performance might because the proposed method based on BERT can only take by most 512 tokens at once (i.e. the maximum number of tokens, L , is 512), when the sequence is longer than 512, the exceeded words will be truncated before being conveyed to the network, making the information incomplete for answer prediction.

To overcome this limitation, the irrelevant part of the subtitles is pruned using similarities between their TF-IDF sentence representations by following (Clark & Gardner, 2018). The pruning is implemented by three steps. First, generate a vocabulary in the TVQA dataset. The vocabulary consists all the tokens that appeared in the TVQA training set for no less than 5 times, which has about 44,000 words. Then, take the full-length subtitles as inputs and segment them every 400 tokens to form different subtitle sections. Finally, compute the cosine similarity between the TF-IDF representations of each section and the question, and select the section with the highest cosine similarity as the input subtitle. The

best validation accuracy of the proposed method with pruning is about 1.1% higher than without pruning, however, the result on the test server is still 4.05% lower than PAMN (J. Kim et al., 2019) and 6.95% lower than STAGE (Lei et al., 2019).

For this phenomenon, there might be two reasons. First, full-length elements contain longer sentences and contain too many words to be covered in BERT embeddings. In the proposed method, the visual concepts/subtitles along with the question and candidate answers are concatenated together. Once the length of the concatenated sequence is longer than L , the remaining words will be truncated. However, previous work (Lei et al., 2018; J. Kim et al., 2019; Lei et al., 2019) encodes the questions, the answers, the subtitles, and the visual features independently, making it possible for the tokenized input to contain more words. In this setting, more information can be conveyed than the proposed method. Second, in previous methods, attention mechanisms are often adopted to find the correspondence between the visual/language features and the question. However, the proposed method does not use attention mechanisms and could hardly find the corresponding part of visual/subtitle elements related to the question. Even if TF-IDF is used for pruning, there are still about 20% of the pruned tokens with more than 512 words. In TVQA model (Lei et al., 2018), context matching modules are used to build context-aware vectors, which help find the related contexts in visual/language features for the correct answer prediction. In PAMN model (J. Kim et al., 2019), dual memory embedding is adopted to enable pinpointing different temporal parts for each module so that their method can correlate the question and related parts. In STAGE model (Lei et al., 2019), guided attention is applied to match the words in questions/answers to the visual concepts and subtitles for the answer prediction. For future work, the explore of attention mechanisms is required to let the improved model pinpoint the related parts from the input, and improve the accuracy of answer prediction when full-length elements are adopted.

Table 3.6

Statistics of the input sequences on the TVQA test set.*

	Max	Min	Avg	>128	>256	>512
Visual	527	20	135.73	45.11%	3.94%	0.01%
Subtitle	684	18	95.58	15.30%	3.30%	0.75%

3.5.3 Evaluation with Different Sequence Lengths

It is also interesting to know how the value of L has influences on the video-QA accuracy. To make this evaluation clear, some statistics about both inputs are firstly given. In this statistics, the maximum number of words, the minimum number of words, the average number of words, and the percentage of having more than 128, 256, and 512 words of visual and subtitle inputs under time stamp annotated cases are calculated. TVQA test* set is selected in this evaluation and the results are shown in Table 3.6.

From Table 3.6, the visual elements tend to have more words than the subtitles on average. Also, there are more than 45% of the sequences having more than 128 words in visual elements, and more than 15% of the sequences with more than 128 words in subtitles. It is also seen that the percentage of sequence with more than 512 words in both elements is less than 1%. This result shows that most of the sequences are fewer than 256 words, meaning the proposed BERT-based representation, which can take 512 tokens by most, can cover most of the information in them.

Three different L values are taken to evaluate the proposed method on the TVQA dataset test* set with time stamp annotations. The L value is selected to be 128, 256, and 512 respectively. Some experiments are performed to see the accuracy of question answering under them. The results are shown in Table 3.7.

From the table, the test* accuracy of the proposed model for the same input sequence gets improved with the increase of L . The reason is that L implies the amount of information conveyed into the BERT network. When L increases, BERT can receive more information about the question from long sequences and will have a better judgment according to the candidate answers. Therefore, the proposed method tends to make use of the provided information and

Table 3.7

Accuracy results (in %) on the TVQA test set using different L values (128, 256, and 512).*

Input Sequence	128	256	512
[CLS]+V/S+Q+[SEP]+A	72.54	72.82	73.05
[CLS]+V/S+.+Q+[SEP]+A	72.23	72.68	72.79
[CLS]+V/S+[SEP]+Q+[SEP]+A	72.35	72.96	73.15

understand the video deeper when L is longer.

3.6 Qualitative Results

Some examples of successful and unsuccessful predictions in the TVQA dataset are shown in Fig. 3.7 respectively. The performance of making correct answer predictions is compared using the proposed model and the TVQA model (Lei et al., 2018), respectively. In the proposed method, [CLS]+V/S+[SEP]+Q+[SEP]+A rearrangements are being used to give BERT representations. The answers in green are the correct answers (ground truths). The words or the bounding boxes in blue are hints to find the correct prediction. Among these examples, questions 1, 3 and 6 are related to the visual features, while questions 2, 4 and 5 are related to the subtitles.

From the figure, the proposed method gives correct answers to questions 1, 2 and 5, but gives incorrect answers to questions 3, 4 and 6. Now, each case will be discussed in detail.

In question 1, the place where they are talking looks like an office. Even if the time annotated subtitles are long and not related to the answer, the proposed method captures the visual features related to an office in the video scene and gives a correct prediction. In question 2, the answer can be found from the subtitles, where Sheldon explains the difference between comics and comic books. This question spans about 12 seconds and is quite challenging. The proposed method gives the correct prediction according to the subtitles. In question 3, Cameron is sitting at the computer while talking in the video frame, but there are many people in the scene. The proposed method does not include any action features and thus, cannot give a correct prediction to this question. In question







<p>(1) </p> <p>Subtitles: 00:00:21,578 --> 00:00:23,346 I told him he was gonna get us both killed. 00:00:23,413 --> 00:00:26,115 (Castle:)But he only got himself killed. Trying to save you. 00:00:27,685 --> 00:00:30,586 (Beckett:)We have an audio recording of Boothe from that night.</p> <p>Q: Where were Castle and Beckett when they were talking to Trey about Lance's death? a0: In a jail cell a1: In an office a2: In a squad car a3: At the crime scene a4: In the parking lot</p> <p>Film: <i>Castle</i> Time stamp: 21.81-29.82</p> <p>Our prediction: 1 (✓) TVQA prediction: 0 (✗)</p>	<p>(2) </p> <p>Subtitles: 00:00:23,334 --> 00:00:26,464 Oh, hey, could you pick me up a few comics for my nephew's birthday? 00:00:26,629 --> 00:00:28,629 No, I think you mean comic books. 00:00:28,798 --> 00:00:30,678 Comics are feeble attempts at humor... 00:00:30,841 --> 00:00:33,631 (Sheldon:)...featuring talking babies and anthropomorphized pets... 00:00:33,803 --> 00:00:38,053 (Sheldon:)...found traditionally in the optimistically named "funny pages."</p> <p>Q: What does Sheldon explain the difference between after Penny asks for a favor? a0: Nuclear fusion and nuclear fission. a1: Sausage and sausage patties. a2: Comics and Comic books. a3: A yard and a meter. a4: Organic chemistry and inorganic chemistry.</p> <p>Film: <i>The Big Bang Theory</i> Time stamp: 25.84-37.81</p> <p>Our prediction: 2 (✓) TVQA prediction: 3 (✗)</p>
<p>(3) </p> <p>Subtitles: 00:00:08,067 --> 00:00:12,231 (Cameron:)House would let you out of it in a heartbeat. Or he wouldn't, just to jerk me around.</p> <p>Q: Who is sitting at the computer when the group is talking? a0: Cameron a1: Chase a2: Foreman a3: House a4: Cuddy</p> <p>Film: <i>House</i> Time stamp: 9.83-11.17</p> <p>Our prediction: 2 (✗) TVQA prediction: 0 (✓)</p>	<p>(4) </p> <p>Subtitles: 00:00:27,494 --> 00:00:28,924 (Stuart:)I appreciate the offer, 00:00:28,928 --> 00:00:31,898 (Stuart:)but actually your mother already gave me the money. 00:00:33,700 --> 00:00:36,000 (Howard:)What? 00:00:36,002 --> 00:00:38,202 (Stuart:)Yeah. I told her it was too much, 00:00:38,204 --> 00:00:41,304 (Stuart:)but she said she was happy to help out her bubala.</p> <p>Q: Who does Stuart say gave him money to reopen his comic book store after Howard offers him some? a0: He inherited some money from a relative a1: Stuart's mom a2: Sheldon a3: Leonard a4: Howard's mom</p> <p>Film: <i>The Big Bang Theory</i> Time stamp: 28.07-41.19</p> <p>Our prediction: 1 (✗) TVQA prediction: 4 (✓)</p>
<p>(5) </p> <p>Subtitles: 00:00:26,373 --> 00:00:28,568 (Masters:)If it's hit her brain, that could mean she doesn't have long. 00:00:28,676 --> 00:00:32,203 So then the question becomes, "Will the sister show up at the funeral?" 00:00:32,279 --> 00:00:33,541 (Chase:)She tried to reconcile. 00:00:33,614 --> 00:00:36,242 (Chase:)But the bad memories crowded out the good and she ran. 00:00:36,350 --> 00:00:37,749 (House:)Nobody's perfect.</p> <p>Q: What reason did Chase give for the patient's sister not talking to her when talking to House on the phone? a0: The sister is actually her mother a1: She didn't remember her a2: They hate each other a3: She doesn't have a sister a4: Bad memories</p> <p>Film: <i>House</i> Time stamp: 27.6-36.5</p> <p>Our prediction: 4 (✓) TVQA prediction: 4 (✓)</p>	<p>(6) </p> <p>Subtitles: 00:00:00,222 --> 00:00:03,862 (Howard:)...before my dad left me and my mom... 00:00:03,859 --> 00:00:07,399 (Howard:)he used to... take me to the comic book store. 00:00:08,463 --> 00:00:11,963 (Howard:)It was one of the few things we did together. 00:00:11,967 --> 00:00:14,837 (Bernadette:)Oh. Howie, I had no idea.</p> <p>Q: How did Bernadette feel when Howard told about his dad? a0: Happy a1: Sad a2: Nervous a3: Anxious a4: Angry</p> <p>Film: <i>The Big Bang Theory</i> Time stamp: 0-14.18</p> <p>Our prediction: 3 (✗) TVQA prediction: 3 (✗)</p>

Figure 3.7. Successful and unsuccessful predictions of the proposed model and TVQA model. Note that the bounding boxes or words in blue are hints to the answer, which are annotated manually.

4, Stuart says in the subtitles that "your mother already gave me the money", then Howard says "What?". The proposed method also fails to predict the correct answer to this question. The reason might be the proposed method finds "Stuart" and "mother" in the same line, but it does not know the relationship between different people, and cannot tell who is the person that gives the money. In question 5, the answer can be directly found from the subtitles, saying "But the bad memories crowded out the good and she ran". The proposed method successfully understands the subtitle and successfully predicts the answer. In question 6, Bernadette's face shows a sad expression, which is very challenging to capture using the visual concepts extracted by Faster RCNN, leading the proposed method predicts incorrectly to this question.

From these examples, the proposed method is able to solve questions related to both visual and language features that cannot be solved by LSTM. However, when the answer is not explicit in either the video frames or the subtitles and needs outside knowledge sources for further information, the proposed method gives a bad prediction. Also, as those visual concept features contain only adjectives and nouns, the proposed method could hardly predict questions related to actions and facial expressions, either. The reason for BERT representation gives better answer predictions in video-QA task may be that BERT uses a self-attention bi-directional structure, which makes every word attend its context on both sides, while in LSTM networks, the follow-up words in the long sentence may have a weak attendance to its long previous words. However, the attention mechanism in TVQA model helps pinpoint the relevant words from the inputs, which might be the main reason for making correct answer predictions to questions 3 and 4.

3.7 Summary

This chapter focused on video question answering task and presented to improve its feature representations to achieve the goal of deeper video understanding. It presented a method that aimed to improve the accuracy of this task by improving the feature representation of the input. The method first retrieved visual

concept features with faster RCNN pre-trained on the Visual Genome dataset, then adopted BERT to give representations to visual concept features and subtitles to capture the semantics from both the video scenes and subtitles more accurately.

Experiments were conducted to test the performance of the proposed method using two different datasets. Different input arrangements, subtitles with (or without) time stamp annotations, and different maximum lengths of the input sequences were also into the proposed method. Results showed that the proposed model understood the semantics in the video frames and subtitles and gave correct answer predictions to the questions based on the TVQA dataset and the Pororo dataset. It improved the accuracy of question answering by 5.09% compared to the previous work based on LSTM network, and 3.34% compared with the STAGE network that used BERT for subtitles, while kept using LSTM for visual feature representation. This means that the proposed method gave a better feature representation to the input elements, it answered the questions related to video contents with higher accuracy and hence, achieved the goal of promoting deeper video understanding.

However, there are still some limitations to the proposed model. First, when the length of input elements is quite long, the accuracy of question answering will drop because the long information will be truncated before being conveyed to BERT network. To solve this problem, the use of attention mechanism might be explored to identify the relevant part of the long subtitles. Second, the proposed model can hardly predict the correct answer to questions that are related to action or facial expressions. The reason might be that the visual concept features contain only adjectives and nouns, but they provide nothing to describe the actions and facial expressions in the video frames. It is required to combine action features and facial expression features along with object features in the visual inputs to broaden the applications of the proposed method. Third, the proposed method does not involve any outside knowledge, yet nothing has been done to let the system know how to reasoning. This makes the prediction of implicit questions difficult. One of the solutions is to import knowledge bases and retrieve the knowledge features along with features from other input elements.

Also, the proposed method should be improved to know how to reason in future studies to find the answer easier.

Chapter 4

A Comparative Study of Language Transformers for Video Question Answering

This chapter, as explicated in Section 1.7, presents a deeper investigation of Transformers for feature representation in video question answering task. It first gives brief introductions to different Transformers, then presents the proposed method, which applies four kinds of Transformers to give representations to the input features in video question answering task. Finally, the best-performed Transformer is found via experiments and the mechanism of Transformers is also discussed.

4.1 Introduction

Video question answering (video-QA) is a task that gives correct answers to the questions related to videos. As video is a media that has multiple modalities, including a sequence of video frames, audio tracks, and speech contents (like subtitles), video-QA systems need to represent the features from these modalities and combine them together. The system will then try to understand what is happening in videos in order to give correct answers to questions.

Recently, video-QA has become a popular research field. Some techniques have been proposed to handle video-QA tasks. The inputs of video-QA vary by task, e.g. video only (Y. Yu, Ko, Choi, & Kim, 2017; J. Gao et al., 2018), video

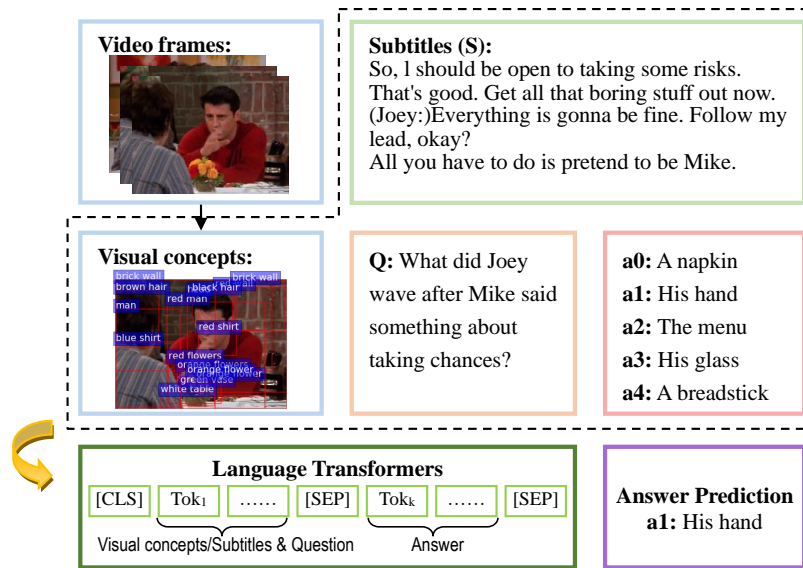


Figure 4.1. Overview of the research. Variants of Transformers are used to encode the complex semantics from video clips in video question answering task.

and subtitle (K.-M. Kim et al., 2018; Lei et al., 2018, 2019), video and audio (Mamgai, Brodiya, Yadav, & Dua, 2019), etc. In this research, the video frame sequences, the corresponding subtitles, a question, and candidate answers are taken as inputs, which is shown in Fig. 4.1. The main motivation for selecting these inputs is that many similar tasks also use video and subtitles, which contain rich information for question answering and have shown outstanding results in previous work (K.-M. Kim et al., 2018; Lei et al., 2018, 2019).

In video-QA models, visual and language elements are usually handled by individual flows (Lei et al., 2018; K.-M. Kim et al., 2017; Tapaswi et al., 2016b). The visual flow retrieves and processes the semantics from video frames, and the language flow obtains and processes the semantics from subtitles in parallel. Recurrent neural networks (RNNs), especially long short-term memory (LSTM) networks (Hochreiter & Schmidhuber, 1997), are used in each flow to encode those elements. As the inputs of video-QA tasks often contain long sequences, RNN-based networks may have difficulty grasping the semantics in them and can thus be a bottleneck to the improvement of accuracy in the video-QA task.

In the natural language processing community, Transformer (Vaswani et al., 2017), a novel architecture that aims at solving sequence-to-sequence tasks while handling long-range dependencies, has been proposed and used extensively (Burns, Tan, Saenko, Sclaroff, & Plummer, 2019; Z. Li et al., 2019; Dasigi, Liu, Marasovic, Smith, & Gardner, 2019; Di Gangi, Negri, & Turchi, 2019). Transformers rely on self-attention to compute representations of the input and output without using RNNs. All of them have been reported to outperform RNNs in several natural language processing tasks.

This work presents a new method to improve the performance of video-QA task. The method adopts pre-trained language Transformer (Vaswani et al., 2017) to capture the visual and language semantics from video clips. It also follows previous work (Lei et al., 2018) and uses a visual concept detector by faster RCNN (Anderson et al., 2018) to represent each detected visual concept with a word or phrase. The network is fine-tuned in an end-to-end manner (including the pre-trained Transformer) for the video-QA task.

This work shares a similar idea to Zeng (Zeng et al., 2017) because both of us are handling visual elements linguistically. However, this work is distinct from Zeng (Zeng et al., 2017) for two reasons: 1) This work uses a sequence of words/phrases to represent a sequence of frames, which is much simpler than the sequences adopted in (Zeng et al., 2017). 2) Besides taking visual concept features, this work also includes the subtitles that contain the transcription of what the characters are saying, which acts complementary to the video frames.

This work is also different from previous work in Chapter 3, which only used BERT for visual and language representations. It was not clear which Transformer performs the best in video-QA task, and why it performs the best before carrying on this work. By evaluating the performance in the video-QA tasks, this work gives a deeper investigation of Transformers. It not only shows the performance of other three Transformers in video-QA task, but also empirically shows how different pre-training strategies and configurations of Transformers affect the video-QA performance.

The contribution of this work is three-fold:

- 1) An improved model for video-QA is proposed. The model uses a pre-

trained language Transformer to encode a sequence of visual concepts besides subtitles.

2) Experiments are performed with four commonly applied Transformers and two video-QA datasets. The performance of different Transformers are evaluated respectively.

3) Discussions are given to analyze how different pre-training strategies and configurations of Transformers affect the video-QA performance.

4.2 Prior Work

In this section, prior work in two aspects is shown: Language Transformers and its applications.

4.2.1 Language Transformers

Before the introduction of Transformers, the method commonly applied in feature representation is mainly based on RNN. Transformer is a novel architecture proposed by Vaswani et al (Vaswani et al., 2017). It relies on self-attention to compute representations of the input and output without using RNNs. Many representative language Transformers have been proposed in these years. For example, Devlin et al (Devlin et al., 2018) propose BERT, which gives deep bidirectional representations of unlabeled text; Dai et al (Dai et al., 2019) introduce transformer-XL, which enables learning dependency beyond a fixed-length sequence without disrupting temporal coherence; Yang et al (Z. Yang et al., 2019) introduce XLNet, which enables bidirectional context learning by maximizing the expected likelihood over all permutations of the factorization order; Liu et al (Y. Liu et al., 2019) present RoBERTa, which gives an improved recipe for training BERT models; and Lan et al (Lan et al., 2019) propose ALBERT to lower the memory consumption and increases the training speed of BERT. On the other hand, Transformers are also applied to combine both visual and language features, like VideoBERT (Sun, Myers, Vondrick, Murphy, & Schmid, 2019), VLBERT (Su et al., 2019) and ViLBERT (J. Lu et al., 2019).

4.2.2 The Applications of Language Transformers

Compared with RNN based methods, Transformers handle long sequences easily. Hence, it is suitable for language feature representation. Lots of work has been proposed to use Transformers for language representation. For example, Lei et al (Lei et al., 2019) propose to use BERT to represent language features in video-QA tasks, they also use RoBERTa for language feature representations in another task related to video moment retrieval (Lei, Yu, Berg, & Bansal, 2020b). Lu et al (L. Lu, Liu, Li, & Gong, 2020) propose to use Transformers in speech understanding; Saffar (Saffar Mehrjardi, 2019) uses Transformers for dialogue generation. Some other applications, like machine translation (J. Zhu et al., 2020), text classification (Z. Xu, 2021), humor prediction (Weller & Seppi, 2019) and social media information retrieval (Murarka, Radhakrishnan, & Ravichandran, 2020).

4.3 Introduction to Transformers

Transformers (Vaswani et al., 2017) are novel architectures without using RNNs. They rely on self-attention to compute the representations of input and output sentences. Compared with RNNs that encode the input sentence word by word, Transformers treat the sentence as a whole (See Fig. 4.2). Transformers transform one sequence into another by reducing the complexity of the algorithm to constant level, so that longer sequences can be processed more easily. Many variants of Transformers have been proposed in recent years in order to represent language features better.

Transformers convert the language features into vectors by input representations, so that computer systems can understand and process them. For a given word token sequence, the input representation consists of the corresponding token embeddings, segment embeddings, and position embeddings. Token embeddings denote the embeddings for the input tokens, segment embeddings denote the sentence that each token belongs to (e.g. A: the former sentence; B: the latter sentence), and position embeddings denote the position of each token within the input sequence. In token embeddings, the first token in every

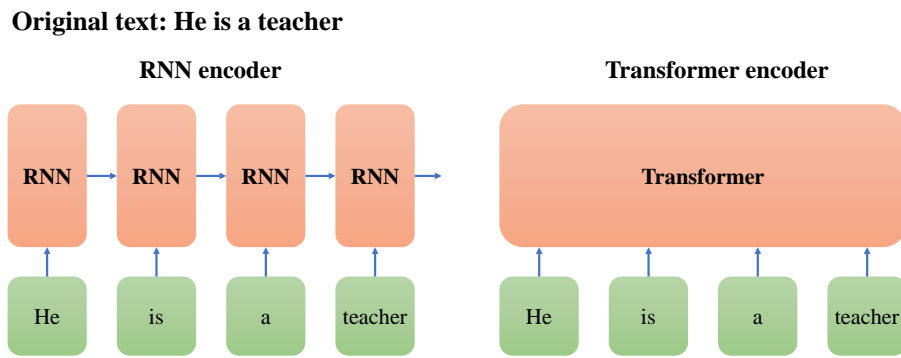


Figure 4.2. RNNs that encode the input sentence word by word, while Transformers treat the sentence as a whole.

sentence is a [CLS] mark, which is used to obtain the output in classification tasks. Then, the former sentence follows. The [SEP] token is added between the former and latter sentences and at the end of the latter sentence to indicate separation between different sentences. The latter sentence then goes between two [SEP] tokens.

It is interesting to know which Transformer performs the best in question answering. Investigations show that the SQuAD task (Rajpurkar, Jia, & Liang, 2018) is similar to the selected task because both tasks need to understand a long series in order to answer questions. When this research is carried, the best performing Transformers on the SQuAD dataset (Rajpurkar et al., 2018), and find that BERT (Devlin et al., 2018), XLNet (Z. Yang et al., 2019), RoBERTa (Y. Liu et al., 2019), and ALBERT (Lan et al., 2019). These Transformers are mainly distinguished from each other by their pre-training settings. In this research, Transformers are used to model the semantics captured from video clips, including visual concept features and subtitles. In the following part, the selected Transformers will be introduced at first, then their performances are going to be evaluated via experiments.

4.3.1 BERT

BERT is a model designed to extract pre-trained deep bidirectional language representations (Devlin et al., 2018). It uses bidirectional Transformers (Vaswani

et al., 2017), meaning every word attends to the context of both sides in every layer of the network. Pre-trained BERT representations can be fine-tuned to achieve state-of-the-art performance in a wide range of tasks (Devlin et al., 2018; Z. Li et al., 2019; Gong et al., 2019).

During pre-training, BERT uses masked language modelling in order to obtain a deep bidirectional representation for the input sequence. Masked language modelling masks some of the input tokens at random, and then predicts those masked tokens. To let the prediction understand the relations between two sentences, the next sentence prediction mechanism is also taken, which gives a binarized prediction of whether a certain sentence is the next sentence of another sentence.

4.3.2 XLNet

XLNet is a generalized method that leverages both auto-regressive language modelling and auto-encoding (Z. Yang et al., 2019). It integrates the segment recurrence and relative encoding scheme of Transformer-XL (Dai et al., 2019) into pre-training and improves the performance for tasks with longer text sequences. Experiments show that XLNet also has good performance on a wide range of tasks (Z. Yang et al., 2019; Dasigi et al., 2019; Zhong et al., 2019).

Instead of masked language modelling, XLNet uses permutation language modelling during pre-training, which learns to utilize contextual features from all positions to capture bidirectional contexts. Next sentence prediction mechanism is removed because it has almost no influence on the prediction.

4.3.3 RoBERTa

RoBERTa is an improved recipe for training BERT models by measuring the impact of hyper-parameters and data size (Y. Liu et al., 2019). Compared with standard BERT, pre-trained RoBERTa model is created by longer training with bigger batches over more data. RoBERTa can match or exceed the performance of BERT and other similar methods (Y. Liu et al., 2019).

During the pre-training, dynamic masking is used. Unlike BERT which per-

forms masking during data pre-processing, RoBERTa generates the masking pattern when a sequence is fed into the model. Next sentence prediction is removed in the RoBERTa pre-training.

4.3.4 ALBERT

ALBERT is proposed as a language representation method (Lan et al., 2019) to reduce memory consumption and training time. To achieve this goal, it has fewer parameters than BERT-large model. Experiments show that ALBERT achieves significantly better performance (Lan et al., 2019) than previous studies.

To reduce the parameters, ALBERT employs factorized embedding parameterization and cross-layer parameter sharing mechanisms during pre-training. In factorized embedding parameterization, the larger embeddings are decomposed into two smaller matrices and projected into a lower-dimensional embedding space. Then, the lower-dimensional space is projected to the hidden layers. In cross-layer parameter sharing, all parameters are shared across layers to improve the efficiencies of parameters.

Instead of next sentence prediction, ALBERT uses sentence order prediction, which focuses on modelling the tie between different sentences. Hence, the ALBERT model learns finer-grained distinctions about discourse-level coherence properties, and consequently, improves the performance for multi-sentence encoding tasks.

4.4 The Proposed Method

The flow chart of the proposed method to answer multiple-choice questions about video is shown in Fig. 4.3. Generally, this model is the same as the one that was proposed in Chapter 3, where two flows for visual and language predictions work respectively at first, and their results are then summed to obtain the joint answer prediction. The visual semantics from each video frame is represented as visual concept features, and the language semantics from the subtitles are directly taken as the input. What is different from previous work

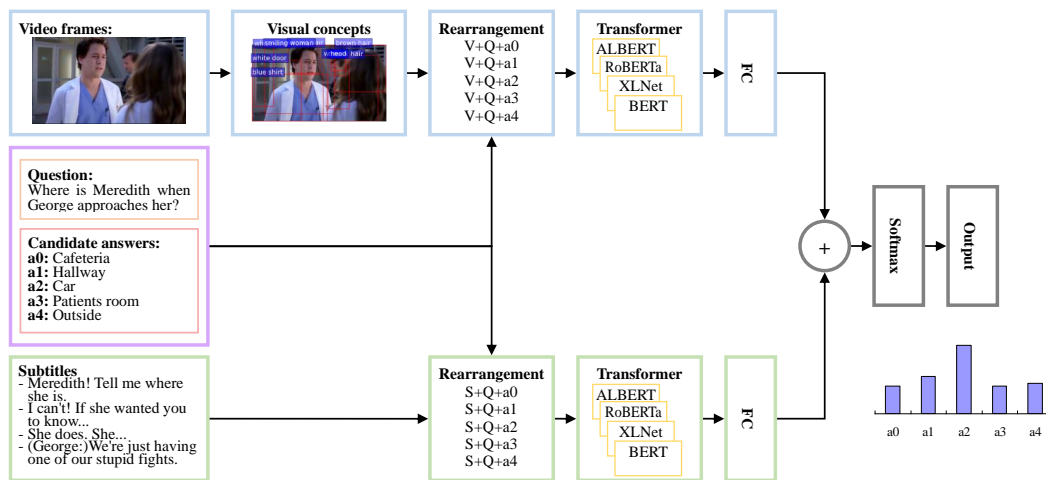


Figure 4.3. Proposed model for answering multiple choice video questions. Note that V, Q, and S denote visual concept, question, and subtitle, respectively.

in Chapter 3 is that this work not only uses BERT to model the visual concept features or subtitles along with the question and each candidate answer in each flow, but also considers other kinds of Transformers. In this work, the same variant of language Transformer but with different instances is used in different flows.

4.4.1 Visual Representations

In the proposed model, visual concept features, which are the labels of the detected objects using Faster-RCNN (Anderson et al., 2018), are used to represent the content of a video scene semantically. The visual concept features are extracted from each video frame using Faster-RCNN (Anderson et al., 2018) pre-trained on the Visual Genome dataset (Krishna et al., 2017) following previous work in Chapter 3. The video frames are extracted at 3 fps. In every extracted frame, the visual concept features are represented by corresponding nouns or phrases, such as *man*, *black hair*, and *blue shirt*, etc. The visual concepts from all the frames are aggregated. Duplicated features are then removed to obtain unique visual concept features from a whole scene. Next, the unique visual concept features v , the question q , and each candidate answer item a_i ($i=0,1,2,3,4$) are concatenated and rearranged into a single string c_i . Each rearranged string

is tokenized to obtain the sequence T_{c_i} .

$$c_i = [v, q, a_i] \quad (4.1)$$

$$T_{c_i} = \text{tokenize}(c_i) \quad (4.2)$$

Here, the concatenation of v and q , $[v, q]$, is set as the former sentence for the Transformer, and $[a_i]$ is set as the latter sentence. When c_i is longer than the maximum number of tokens L , the numbers of token in the former ($[v, q]$) and latter ($[a_i]$) sentences are measured. The last token(s) will be truncated from the longer sentence until the number of words in T_{c_i} is no more than L .

Next, T_{c_i} is fed into the language Transformers to obtain an output V_{c_i} , a matrix containing the vector representation of each word in the input sentence. The output vector corresponding to the [CLS] token, $V_{c_i}^0$, is fed into a fully connected layer to obtain the visual flow representation R_{c_i} for answer a_i , where F_c is a trainable parameter.

$$V_{c_i} = \text{Transformer}_c(T_{c_i}) \quad (4.3)$$

$$R_{c_i} = F_c V_{c_i}^0 \quad (4.4)$$

4.4.2 Language Representations

Similarly, in the language flow, the subtitles s , the question q , and the candidate answer items a_i ($i=0,1,2,3,4$) are concatenated to form the string w_i . The concatenation of s and q , $[s, q]$, is set as the former sentence, and $[a_i]$ is set as the latter sentence. The last token(s) will be truncated from the longer sentence until the number of words in T_{w_i} is no more than L . The rearranged string is tokenized to form the sequence of tokens T_{w_i} , which is fed into the Transformer to obtain V_{w_i} . Then, $V_{w_i}^0$ is fed into a fully connected layer to obtain the language

flow representation R_{w_i} for answer a_i , where F_w is a trainable parameter.

$$w_i = [s, q, a_i] \quad (4.5)$$

$$T_{w_i} = \text{tokenize}(w_i) \quad (4.6)$$

$$V_{w_i} = \text{Transformer}_w(T_{w_i}) \quad (4.7)$$

$$R_{w_i} = F_w V_{w_i}^0 \quad (4.8)$$

4.4.3 Prediction

Finally, the representations of visual and language flows for each question are summed to obtain R_{p_i} . Softmax is used to convert the summed vector into the answer scores R_f :

$$R_{p_i} = R_{c_i} + R_{w_i} \quad (4.9)$$

$$R_p = [R_{p_0}, R_{p_1}, R_{p_2}, R_{p_3}, R_{p_4}] \quad (4.10)$$

$$R_f = \text{softmax}(R_p) \quad (4.11)$$

The answer with the maximum score is selected as the final predicted answer a_p with:

$$a_p = \text{argmax}(R_f) \quad (4.12)$$

4.5 Experiments

The proposed method is implemented based on Python 3.7 and Pytorch. The training parameters in the experiments are shown in Table 4.1. The detailed in-

Table 4.1

The training parameters in the experiment.

Parameters	Setting
Learning Rate	2×10^{-5}
Number of Epoch	10
Training Batch Size	8
Inferring Batch Size	16
Max Number of Tokens (L)	128
Optimizer	Adam
Weight Decay	1×10^{-5}

Table 4.2

Detailed information about different pre-trained Transformer models.

Transformer	Model name	Layers	Hidden size	Attention heads	Parameters
BERT	bert-base-uncased	12	768	12	110M
XLNet	xlnet-base-cased	12	768	12	110M
RoBERTa	roberta-base	12	768	12	125M
ALBERT	albert-base-v1	12	768	12	11M

formation about the selected pre-trained Transformer models is shown in Table 4.2. The accuracy of question answering is taken as the evaluation metric in the experiments. Cross-entropy loss is applied in the prediction.

4.5.1 Experimental Setting

Datasets The experiments follow the previous chapter, and use the TVQA dataset (Lei et al., 2018) and the Pororo dataset (K.-M. Kim et al., 2017). The TVQA dataset is a video-QA dataset based on 6 popular TV shows, and the Pororo dataset is based on a children’s Cartoon series called *Pororo*. Each question in both dataset is designed to have multiple-choice style with one correct answer out of five candidate answers. For each question, subtitles corresponding to each video scene are also provided. To correctly answer these questions, a joint understanding is required to both visual and language features. Results on TVQA (Lei et al., 2018), STAGE (Lei et al., 2019), and MDAM (K.-M. Kim et al., 2018) are reported to compare the results of the proposed model with those of previous models.

Input Sequence To test the performance of Transformers, the input tokens with visual and language features are rearranged in three different ways:

$$\begin{aligned}
 &1) V/S + Q + A \\
 &2) V/S + . + Q + A \\
 &3) V/S + [\text{SEP}] + Q + A
 \end{aligned} \tag{4.13}$$

Where V represents visual concepts, which corresponds to v in Section 4.4, S represents subtitles, which corresponds to s , Q represents question, which corresponds to q , and A represents answer, which combines all a_i . V/S indicates that both visual concepts and subtitles are taken in visual and language flows. Ablation studies are conducted by removing either visual concepts ($S+Q+A$) or subtitles ($V+Q+A$). The highest accuracy of each Transformer is reported along with its corresponding word rearrangements.

4.5.2 Results on the TVQA Dataset

Results with time stamp annotated elements: The experiments are first performed on the TVQA dataset with time stamp annotated elements. The results are shown in Table 4.3. The TVQA dataset has no answer label in the test set, so some of the test set predictions are submitted to the test server. As the number of total submissions to the test server is limited, a test* set is created by splitting 15,253 Q/A pairs from the training set, while the validation set is kept the same. For comparison, some results of previously-proposed methods are also reported, including the TVQA model (Lei et al., 2018), which uses a shared LSTM for both visual and language representations and two unique LSTMs for joint-modelling, the results from Thomas et al (Winterbottom, Xiao, McLean, & Al Moubayed, 2020), which takes the TVQA model’s framework, but uses either shared LSTM and BERT for comparison and enlarges the output dimension of the unique LSTM; and STAGE (Lei et al., 2019), which uses LSTM for visual and BERT for language representation. Also, some of the answer predictions are submitted to the server to obtain their results on the official test set.

Table 4.3

Accuracy (in %) of proposed model and baselines on TVQA dataset with time stamp annotations. Note that one only has limited chances to submit their results to the test server for evaluation; thus, only some representative results of the proposed model are shown to the public.

Input	Name	Model	Rearrangements	Val	Test*	Test
V+Q+A	TVQA (Lei et al., 2018)	GloVe + LSTM	-	45.03	-	45.44
	Thomas (Winterbottom et al., 2020)	GloVe + LSTM	-	45.39	-	-
	Thomas (Winterbottom et al., 2020)	BERT	-	43.44	-	-
	Proposed	BERT	V+.+Q+A	48.95	49.23	-
	Proposed	XLNet	V+.+Q+A	48.17	48.53	-
	Proposed	RoBERTa	V+Q+A	27.99	27.66	-
	Proposed	ALBERT	V+Q+A	47.01	47.53	-
S+Q+A	TVQA (Lei et al., 2018)	GloVe + LSTM	-	65.15	-	66.36
	Thomas (Winterbottom et al., 2020)	GloVe + LSTM	-	66.07	-	-
	Thomas (Winterbottom et al., 2020)	BERT	-	68.30	-	-
	Proposed	BERT	S+[SEP]+Q+A	70.65	70.22	-
	Proposed	XLNet	S+.+Q+A	68.21	68.22	-
	Proposed	RoBERTa	S+.+Q+A	69.32	68.92	-
	Proposed	ALBERT	S+[SEP]+Q+A	67.53	66.57	-
V/S+Q+A	TVQA (Lei et al., 2018)	GloVe + LSTM	-	67.70	-	68.48
	STAGE (Lei et al., 2019)	LSTM + BERT	-	70.50	-	70.23
	Proposed	BERT	V/S+.+Q+A	72.41	72.23	72.71
	Proposed	XLNet	V/S+[SEP]+Q+A	70.28	70.33	70.64
	Proposed	RoBERTa	V/S+.+Q+A	69.09	69.11	68.82
	Proposed	ALBERT	V/S+.+Q+A	70.32	69.65	70.35

From the results in the test set, the accuracy of question answering using Transformers are all better than the TVQA model when V/S+Q+A are taken as inputs. Generally, the results are even better than STAGE. However, when RoBERTa is taken for V+Q+A inputs, the validation accuracy is 17.04% lower than TVQA baseline and is 17.40% lower than Thomas et al. Among these Transformers, BERT is the best performer in answer predictions: Results show that when both visual and subtitle representations are taken, the proposed model using BERT obtains an accuracy up to 4.23% higher than that TVQA baseline and up to 2.48% higher than STAGE.

It can also be found that the results using V+Q+A is about 20% lower than using S+Q+A, while the results using S+Q+A are close to using V/S+Q+A. The reason might lie in the bias in the TVQA dataset. For this issue, Thomas et al (Winterbottom et al., 2020) have proved that TVQA dataset tends to obtain more information from the subtitles, while suppress the information in the video during training. Hence, once the language representation is strong enough,

the accuracy will improve a lot. As both methods (i.e. LSTM-based methods and Transformer-based methods) in Table 4.3 are natural language processing frameworks, this bias-related finding explains why the prediction of subtitle-based questions is better.

BERT is not the best performer among the four variants of Transformers in SQuAD, but it is the best in the experiments of the proposed method on the TVQA dataset. To study the reason why BERT performs the best here, it is important to consider the different pre-training settings in different Transformers, which mainly refers to the language modelling strategy in each Transformer.

The language modelling strategies in different Transformers influence the way to predict answers according to visual and language features. When modelling language features, Transformers will temporarily *cover* some words in the input sentence, and try to predict them based on the nearby contexts. In the proposed model, most of the answers are implied in the given context (V/S). When BERT is adopted for feature representations, the masked language modelling masks some words in the sentences randomly and uses the contexts on both sides to predict that word during pre-training. In XLNet, when the sentence is short, there are fewer permutations for words, giving only limited information for the permutation language modelling to predict the words while keeping its long memory. The proposed model takes no more than 128 tokens in training, which is only a quarter of that in the experiments with XLNet baseline (512). The information for word prediction becomes limited and the performance is not as good as BERT's. When RoBERTa is used, the masking pattern is generated when a sequence is fed into the model. This means RoBERTa also needs to fine-tune the masks and fit the context structure while training. Because SQuAD is a reading comprehension task, the input in SQuAD is a paragraph with several sentences, the tie between different sentences is strong, and the masks fit the context structure well. However, the subtitles in the video are mainly dialogues, the tie between sentences is weaker than between paragraphs, and the masks cannot fit the context structure as well as SQuAD, making the performance of RoBERTa insufficient. In ALBERT, large embeddings are factorized into smaller ones, which breaks some word embeddings and cannot lead to a

better prediction compared with BERT.

When V+Q+A is taken as input, the accuracy of RoBERTa is much lower than that of other Transformers. This is also because of the dynamic masking in RoBERTa. When RoBERTa is used for feature representation, it needs to know how to predict the words in the masks during training. By using subtitles and questions, dynamic masking can learn from a certain structure of the input sentence, so the accuracy of answer prediction is only slightly lower than that of BERT. However, when visual concept features, which are just word sequences, are adopted, the dynamic masking cannot learn from a certain structure of the input sentence and gives many wrong predictions.

Next, the function of next sentence prediction is also studied in the proposed method. The next sentence prediction mechanism is designed to find the next sentence (A) according to the previous sentences (V/S+Q), and hence, might be responsible for the drop of accuracy in V+Q+A when RoBERTa is used for feature representation. It can be found that only BERT has such a mechanism (XLNet and RoBERTa do not), while ALBERT has improved next sentence prediction to sentence order prediction. Previous work (Z. Yang et al., 2019; Y. Liu et al., 2019; Lan et al., 2019) has shown that next sentence prediction is almost ineffective because it lacks difficulty compared with language modelling. To check the results, the performance of XLNet is close to that of BERT, meaning that next sentence prediction is not the main reason for such a drop.

Results with Full-Length Elements: It is challenging to test the performance of the proposed method with full-length visual concept features and subtitles. For this evaluation, full-length elements (i.e. visual concepts and subtitles without time stamp annotations) are used instead of the time stamp annotated elements in the TVQA dataset. Different from previous work in Chapter 3, L , the maximum number of tokens per input, is still set as 128. Some pre-processing techniques are conducted before the data being taken as the input, because full-length elements contain many tokens and some of them may be truncated before being conveyed into the proposed model directly. In this work, an 80-token, 10-step sliding window (i.e. There are a maximum of 80 tokens in each segment, and it will move by 10 tokens to obtain the next segment) is used in the lan-

Table 4.4

Accuracy (in %) of proposed model and baselines using full length elements on TVQA dataset.

Input	Name	Model	Rearrangements	Val	Test
V/S+Q+A	TVQA (Lei et al., 2018)	LSTM	-	64.42	66.46
	PAMN (J. Kim et al., 2019)	LSTM	-	-	66.77
	STAGE (Lei et al., 2019)	LSTM+BERT	-	68.56	69.67
	HERO (L. Li et al., 2020)	Transformers	-	-	71.25
	Proposed	BERT	V/S+.+Q+A	65.07	66.44
	Proposed	XLNet	V/S+[SEP]+Q+A	61.40	61.68
	Proposed	RoBERTa	V/S+.+Q+A	63.86	64.23
	Proposed	ALBERT	V/S+.+Q+A	63.69	63.56

guage flow to select the tokens that best match the question and five candidate answers. The questions and five candidate answers are concatenated together and the stop words are removed from them so that the model can concentrate more on the main components of the question and answers. The cosine similarity of the TF-IDF index between the tokens in the window and question-answer pair is calculated, and used as a metric for selecting the input tokens. Note that this method is different from the previous one proposed in Chapter 3, where L is set as 512 and the input token is divided into several segments directly.

In implementation, a vocabulary is firstly generated to cover approximately 44,000 words with all the tokens in the TVQA training set that appear at least 5 times. Then, the tokenized subtitles are segmented into 80-token segments and cosine similarity between the TF-IDF representations of each segment and the question is computed. Finally, the segment with the highest cosine similarity is selected as the input subtitle. The results of each model are listed in Table 4.4.

From the table, the best test accuracy of the proposed model is 0.02% lower than that of the TVQA baseline. There may be two main reasons for this phenomenon. First, full-length elements contain too many words to be covered in the embeddings, because the proposed model embeds the visual concepts/subtitles, the question, and candidate answers altogether and truncates the remaining words when the input sequence is longer than L , while the TVQA model (Lei et al., 2018), STAGE (Lei et al., 2019), PAMN (J. Kim et al., 2019) embeds each feature independently. Second, the proposed model does not use attention to find the corresponding part of the visual/subtitle elements related to

the question. In the TVQA model (Lei et al., 2018), STAGE (Lei et al., 2019), PAMN (J. Kim et al., 2019), and HERO (L. Li et al., 2020), attention mechanism is adopted to help find the corresponding visual frames and subtitles related to the question. In future work, the exploration of attention mechanisms is required to continue improving the prediction in full-length elements.

4.5.3 Results on the Pororo Dataset

Next, the evaluation of the model is carried using the Pororo dataset. The results are compared against the MDAM (K.-M. Kim et al., 2018) model and TVQA (Lei et al., 2018) framework (i.e. The experiments use the model provided by TVQA but change the dataset to Pororo). The results are shown in Table 4.5. On the Pororo dataset, the accuracy of the proposed method is 6.73% higher than that of MDAM and 11.26% higher than that of TVQA. As the Pororo dataset has no time stamp annotations, full-length elements are used directly. The experiment does not make comparison with STAGE (Lei et al., 2019) because their pre-trained BERT model is not available for the Pororo dataset.

From the results, the accuracy of the proposed method using Transformers with visual and language features gets improved compared with those using LSTM. Among these Transformers, it is ALBERT that has the highest accuracy on the Pororo test set. It increases the accuracy by 6.73% compared with MDAM (K.-M. Kim et al., 2018). When RoBERTa is taken for feature representation, the accuracy of using V+Q+A is lower than using other Transformer, and the accuracy of using V/S+Q+A drops by 14.13% approximately compared with using S+Q+A in the test set. The reason for such a drop might lie in the structure of visual concept features.

From the results, the best accuracy of V+Q+A on the Pororo test set using Transformers is still 0.82% lower than MDAM (K.-M. Kim et al., 2018), and the validation accuracy of BERT, RoBERTa, and ALBERT when using V/S+Q+A are lower than those using S+Q+A. The reason is considered to be the representation method of visual features.

In MDAM (K.-M. Kim et al., 2018), the visual features are directly retrieved

Table 4.5

Accuracy (in %) of proposed model and baselines on Pororo dataset.

Input	Name	Model (Rearrangements)	Val	Test
V+Q+A	MDAM (K.-M. Kim et al., 2018)	LSTM	-	42.10
	TVQA (Lei et al., 2018)	GloVe + LSTM	34.87	33.77
	Proposed	BERT (V+Q+A)	40.75	40.03
	Proposed	XLNet (V+Q+A)	38.14	41.28
	Proposed	RoBERTa (V+Q+A)	31.70	35.25
	Proposed	ALBERT (V+.+Q+A)	37.83	38.93
S+Q+A	MDAM (K.-M. Kim et al., 2018)	LSTM	-	42.50
	TVQA (Lei et al., 2018)	GloVe + LSTM	37.60	33.90
	Proposed	BERT (S+[SEP]+Q+A)	52.51	54.16
	Proposed	XLNet (S+Q+A)	50.61	52.24
	Proposed	RoBERTa (S+Q+A)	53.07	54.01
	Proposed	ALBERT (S+.+Q+A)	52.04	53.64
V/S+Q+A	MDAM (K.-M. Kim et al., 2018)	LSTM	-	48.90
	TVQA (Lei et al., 2018)	GloVe + LSTM	37.78	42.53
	Proposed	BERT (V/S+.+Q+A)	52.35	55.26
	Proposed	XLNet (V/S+Q+A)	51.38	54.89
	Proposed	RoBERTa (V/S+[SEP]+Q+A)	37.17	39.88
	Proposed	ALBERT (V/S+.+Q+A)	51.48	55.63

as tensors from ResNet-152 pre-trained on the ImageNet set, while in the proposed model, the visual features are first retrieved as tensors, then classified, and finally represented as words from Faster-RCNN pre-trained on the Visual Genome set. There might be some classification errors in the visual concept features, as Pororo is a cartoon-based dataset, while Visual Genome is a real-world-based image dataset. Such errors make the Faster R-CNN detector not sufficiently accurate to give correct answer predictions.

To illustrate this clearly, some detected features are shown from the video frames in the TVQA dataset and the Pororo dataset respectively in Fig. 4.4, where the features in the top line are from the TVQA dataset, and those in the bottom line are from the Pororo dataset.

From the figure, it can be seen that the detected features in TVQA scene are more than Pororo, and Faster-RCNN detects features in the TVQA dataset accurately. In the Pororo dataset, it detects the photo album pages (bottom left) as *a window*, and the cartoon figure (bottom right) as a *yellow toy*. From

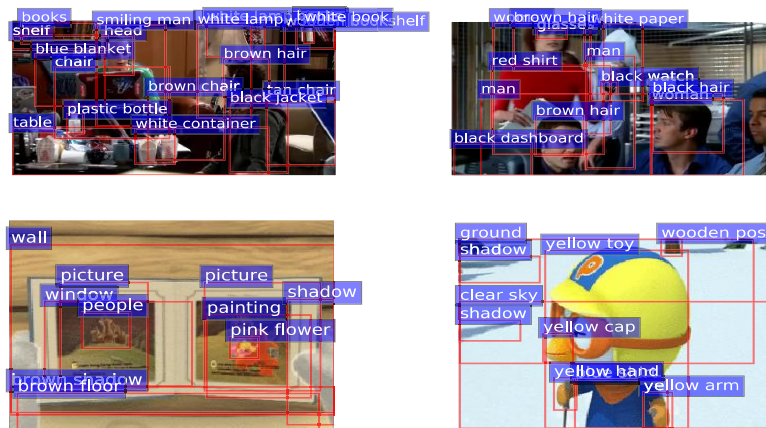


Figure 4.4. Examples of detected features in TVQA and Pororo datasets. The top line shows frames from the TVQA dataset, and the bottom line shows frames from the Pororo dataset.

the figure, Faster-RCNN detector cannot retrieve cartoon features as accurately as real-world features, which should be responsible for the drop in validation accuracy when using V/S+Q+A.

4.6 Discussion

4.6.1 Training Time and Inferring Time

It is important to compare which model runs faster during the training and inferring in this task. Thus, the training time per epoch and estimate the inferring time per question are reported. The inferring time per question is estimated by dividing the total time needed in inferring and the total number of questions to infer. The experiments are performed on a computer with a Core i7 8700K CPU (3.70 GHz), 32G RAM, and a Nvidia TITAN RTX GPU. Other configurations are kept the same while the model is kept being changed. The training time per epoch and the inferring time per question with different Transformers on the TVQA dataset are listed in Table 4.6.

From the table, all the Transformer-based models need more time for training and inferring compared with TVQA baseline (Lei et al., 2018). This means that the Transformers is more time-consuming and memory-consuming. Also, the training time of using V+Q+A is a bit shorter than S+Q+A, while the inferring

Table 4.6

Training time (in h:mm, per epoch) and inferring time (in ms, per question) for different Transformers on the TVQA dataset.

Input	Model	Training time	Inferring time
TVQA (Lei et al., 2018)	V+Q+A	2:14	5.70
BERT		2:57	21.29
XLNet		4:13	30.46
RoBERTa		2:50	18.21
ALBERT		2:42	17.49
TVQA (Lei et al., 2018)	S+Q+A	2:16	5.70
BERT		2:59	21.36
XLNet		4:15	30.46
RoBERTa		2:52	17.95
ALBERT		2:45	17.36
TVQA (Lei et al., 2018)	V/S+Q+A	4:15	10.09
BERT		5:27	32.69
XLNet		7:45	54.77
RoBERTa		5:22	30.01
ALBERT		4:55	33.87

time of using V+Q+A is no shorter than S+Q+A, except XLNet is used. The best validation is usually obtained within 3 epochs in both flows in the proposed model and the TVQA model.

4.6.2 Comparison with High-level Visual Features

To know whether visual concept features perform better than high-level visual features, a comparison is made based on the results of question answering between them. For this comparison, the official test set of the TVQA dataset with time stamp annotation is used. The selected visual features are visual concept features and the high-level features retrieved using ResNet-101 (K. He et al., 2016a) pre-trained on the ImageNet dataset (Deng et al., 2009). The accuracy of question answering is shown in Table 4.7.

From the table, it can be seen that the accuracy of question answering using LSTM with visual concept feature is 1.54% higher than with ResNet-101 features. This means that visual concept features work better than ResNet-101 features, and is more suitable to describe the main contents in the video frames. It is also obvious that the proposed method based on Transformers has better

Table 4.7

Accuracy of question answering using different features (in %).

Input	Name	Visual Features	Model	Test
V/S+Q+A	TVQA (Lei et al., 2018)	ResNet-101	LSTM	66.94
	TVQA (Lei et al., 2018)	Visual concept	LSTM	68.48
	Proposed	Visual concept	BERT	72.71
	Proposed	Visual concept	XLNet	70.64
	Proposed	Visual concept	RoBERTa	68.82
	Proposed	Visual concept	ALBERT	70.35

accuracy than the method based on LSTM, meaning the proposed method understands the semantic deeper in the input elements.

4.6.3 Attention Weights in Different Flows

The attention weights of BERT, the best-performed Transformer on the TVQA dataset, are visualized via BERTviz (Vig, 2019) to give a deeper investigation about the mechanism of answer prediction in Transformers. A question is selected as an example and the attention weights of the visual and language flows are visualized at layers 1, 7, and 12 in Fig. 4.5 respectively. The answer to this question can be found in the language flow. Note that the height of the full attention visualization is very large, so this figure only puts the focus showing the assembling of attention here.

From the figure, the attention weights tend to assemble at the [CLS] mark at layer 1 of both flows, meaning BERT starts by [CLS] for sentence classification. They tend to assemble at the [SEP] mark in layer 7 of both flows, implying BERT is trying to find the correspondence of the former sentence (V/S+Q) and the latter sentence (A). However, the attention weights tend to assemble in different ways the two flows in layer 12: they tend to assemble at the punctuation in the language flow while still at the [SEP] mark at the visual flow. This implies that BERT has found the correspondence between the answer and the input sequence and tends to know the sentence structure in the language flow, but in the visual flow, it could not find correspondence between the answer and the input sequence, and is still *trying* to find it.

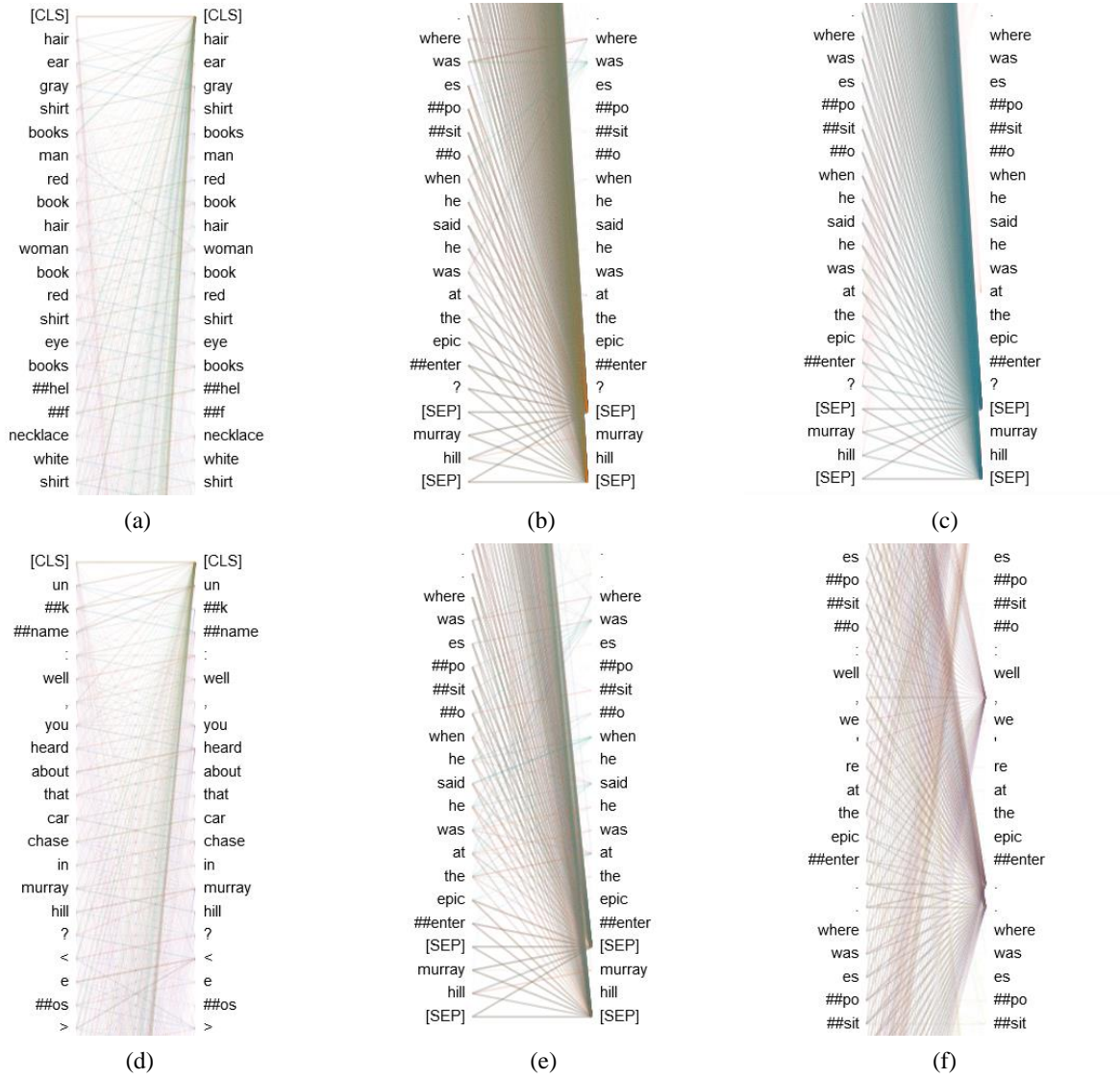


Figure 4.5. Attention weights of the BERT model in the visual and language flow. (a) Visual flow, Layer 1, (b) Visual flow, Layer 7, (c) Visual flow, Layer 12, (d) Language flow, Layer 1, (e) Language flow, Layer 7, (f) Language flow, Layer 12. Note that the height of the full attention visualization is very large, so the figure only puts the focus (i.e. where the attention focuses) here.

Table 4.8

Accuracy results (in %) on the TVQA test set using different L values (128, 256, and 512).*

	BERT	XLNet	RoBERTa	ALBERT
128	72.23	70.33	69.11	69.65
256	72.68	72.45	66.47	70.20
512	72.79	72.98	66.53	70.45

4.6.4 Evaluation with Different Sequence Lengths

The length of each sequence L denotes the amount of information that can be conveyed at once to the neural network. Different L values may have influences on the accuracy of answer predictions. The statistics about the TVQA test* set with time stamp annotations are given in Table 3.6. To evaluate the performance of the proposed model, three different values (128, 256, and 512) of L are taken as the length of input sequence on the TVQA test* set with time stamp annotations using different kinds of Transformers. The results are listed in Table 4.8.

From the results above, even if the accuracy of XLNet is 2.54% lower than that of BERT when $L=128$, when L is increased to 512, the performance of XLNet is slightly higher than that of BERT, meaning XLNet gives better feature representations when it meets longer sequences. For RoBERTa, the performance might be influenced by the visual concept features, which are composed of words or phrases instead of a whole sentence, because 45.11% of visual sequences are more than 128 words, being about three times of that in subtitles. For BERT and ALBERT, the performance is stable regardless of adding or reducing the length of L .


4.6.5 Qualitative Results

Some examples of successful and unsuccessful predictions using different Transformers on the TVQA dataset and the Pororo datasets are shown in Fig. 4.6. Among these examples, the word arrangements for each Transformer in Table 4.5 is used. The answer in green is the correct answer (ground truth). The bounding boxes or the words in blue are hints to the correct predictions, which

are annotated manually for a better demonstration. To categorize by dataset, questions (1) to (4) are from the TVQA dataset, and questions (5) and (6) are from the Pororo dataset. To categorize by type, questions (1), (3), and (5) are related to the visual features, while questions (2), (4), and (6) are related to the subtitles. Note that the subtitles provided to answer questions (4), (5), and (6) are long, so some unrelated parts are cut off. As the original subtitles in the Pororo dataset have no punctuation, some necessary punctuation are also added to make them easy to read.

From the figure, when BERT, XLNet, and ALBERT are adopted for video feature representations, the proposed method gets six, five, and four correct predictions, respectively. When RoBERTa is used, the proposed model only gets two correct predictions. Now, each case will be discussed in detail.


In question (1), the scene is set in a park, and a couple is sitting on a bench. When BERT, XLNet, and ALBERT are used to represent input features, the proposed model captures the visual features related to park and bench in the video scene, so it gives correct answer predictions to this question; RoBERTa and the TVQA baseline cannot deal with the video scene correctly and hence give incorrect answer predictions. In question (2), the answer can be found from the subtitles, where the actor is trying to explain the difference between comics and comic books. This question has a duration of approximately 12 seconds and is quite challenging for the computer to understand. All of the four Transformers give correct predictions according to the subtitles, but the TVQA baseline fails to give a correct prediction. In question (3), beer bottles are on the desk when they are talking in the video frame, BERT, XLNet, and RoBERTa can predict the answer correctly, but ALBERT and the TVQA baseline give wrong answer predictions. In question (4), Chandler says he is officially wiped, and another person says they should *get going* (leave). When BERT, XLNet, and the TVQA baseline are adopted, they successfully find the correspondence between *get going* and *leave* and finally give correct predictions, while RoBERTa and ALBERT cannot find correspondences between them, and thereby generating incorrect predictions. In question (5), although the subtitle is saying *Hold on. I will come save you*, it is difficult to know *Who* is to be saved *tongtong*.

(1) 

Subtitles: (EXHALES) (Eve:)Do you think the guy who did this to me feels bad?	Q: What was House sit on when Eve asked if the guy felt bad? a0: The floor in an office a1: A desk in an office a2: A couch in a living room a3: A park bench a4: The foot of the bed
--	--

Predictions:


BERT	XLNet	RoBERTa	ALBERT	TVQA
a3 (✓)	a3 (✓)	a1 (✗)	a3 (✓)	a2 (✗)

(2) 

Subtitles: Oh, hey, could you pick me up a few comics for my nephew's birthday? <i>No, I think you mean comic books.</i> <i>Comics are feeble attempts at humor...</i> (Sheldon:)...featuring talking babies and anthropomorphized pets... (Sheldon:)...found traditionally in the optimistically named "funny pages."	Q: What does Sheldon explain the difference between after Penny asks for a favor? a0: Nuclear fusion and nuclear fission. a1: Sausage and sausage patties. a2: Comics and Comic books. a3: A yard and a meter. a4: Organic chemistry and inorganic chemistry.
--	--

Predictions:


BERT	XLNet	RoBERTa	ALBERT	TVQA
a2 (✓)	a2 (✓)	a2 (✓)	a2 (✓)	a3 (✗)

(3) 

Subtitles: (Robin:)I did. (Marshall:)And now he's coming here? (Marshall:)Robin, why do you want to see this guy? (Lily:)Oh! I know why. (Lily:)You're gonna win, aren't you? (Robin:)I have no idea what you're talking about. (Ted:)Yes, you do. (Ted:)Whenever you haven't seen someone for a long time,	Q: What drink bottle is at the table when Robin, Lily, Marshall, and Ted are talking to each other? a0: seltzer a1: soda a2: Snapple a3: Iced tea a4: beer
--	---

Predictions:


BERT	XLNet	RoBERTa	ALBERT	TVQA
a4 (✓)	a4 (✓)	a4 (✓)	a2 (✗)	a2 (✗)

(4) 

Subtitles: <i>(Chandler:)This has been great, but I'm officially wiped.</i> <i>Me too. We should get going.</i> No! Come on, you guys! Come on, it's only 11:30. (Rachel:)Who here does not have the time to get to know Julie? (Chandler:)I got the time to get to know Julie.	Q: Who does Stuart say gave him money to reopen his comic book store after Howard offers him some? a0: Chandler wants to leave because he has to make a phone call a1: Chandler wants to leave because he has a date a2: Chandler wants to leave because they are all mad at him a3: Chandler wants to leave because he is hungry a4: Chandler wants to leave because he is tired (wiped)
---	--

Predictions:


BERT	XLNet	RoBERTa	ALBERT	TVQA
a4 (✓)	a4 (✓)	a1 (✗)	a2 (✗)	a4 (✓)

(5) 

Subtitles: Tu tu and Tongtong. Today pororo and his friend are going on a picnic. All right everyone gets on mm hm. OK. Then here we go. Wow! Oh no another branch is about to break Noo tongtong. Tu tu ugh. <i>Hold on I will come save you.</i> Ugh no do not it is dangerous. And that is how Tu tu came to live with tongtong Till we meet again, Tu tu	Q: Who shows up to save tongtong? a0: Crong appears and tries to save him a1: Pororo and petty appear and try to save him a2: Nyao appears and tries to save him a3: Tu tu appears and tries to save him a4: Loopy and eddy appear and try to save him
---	---

Predictions:

BERT	XLNet	RoBERTa	ALBERT
a3 (✓)	a3 (✓)	a1 (✗)	a3 (✓)

(6) 

Subtitles: And the last team is Eddy and Rody. I will show you who the best. Now smile for the picture let me see you smile. Cheese. <i>We almost won some luck.</i> That is called skills not luck. What I won did not I? Really? Of course! Eddy made a very interesting sled. Hope he comes home safe.	Q: Why did eddy say pororo and crong won the first race? a0: Because puroro cheated during the first race a1: Because his sled wasn't good enough a2: Because they were lucky a3: Because crong is lighter a4: Because their path was easier
--	---

Predictions:

BERT	XLNet	RoBERTa	ALBERT
a2 (✓)	a1 (✗)	a4 (✗)	a2 (✓)

Figure 4.6. Successful and unsuccessful predictions on TVQA and Pororo datasets. (1) to (4) are examples from the TVQA dataset, (5) and (6) are examples from the Pororo dataset. Note that the bounding boxes or words in blue are hints to the answer, which are annotated by us manually.

When looking at the video frames, it is *Tutu*. To answer this question correctly, the proposed model needs to understand the relationship between characters, which is not considered before. BERT, XLNet, and ALBERT might make use of the previous sentence (*Tu tu, ugh*) to find the answer. In question (6), the subtitle says they *almost won some luck*, so the answer is *they are lucky*. BERT and ALBERT can connect these two phrases and predict the answer well, while XLNet and RoBERTa cannot connect them and hence give wrong answer predictions to the question.

From these examples, the proposed model is able to have a deeper understanding of videos. It is also able to solve questions related to both visual and language elements that cannot be solved by LSTM. This means the self-attention bidirectional structure in Transformers gives a better representation for both visual and language elements.

However, there are still some limitations to this method. When the answer is not explicit in the video frames or the subtitles and needs reasoning, the proposed model cannot perform it and thus gives bad predictions. When the answer is related to outside knowledge, the proposed model could hardly give prediction to it either. Also, the visual features contain only adjectives and nouns and do not contain any actions. When the question is about actions, the proposed method cannot receive the related information from video frames, and thus, can hardly give a correct answer prediction.

4.7 Summary

This chapter followed the previous chapter to keep studying video question answering task for deeper video understanding. A model was proposed to first retrieve visual concept features from video frames with faster RCNN pre-trained on Visual Genome dataset, and then use Transformers to give feature representations for visual features and subtitles in order to capture the semantics from both the video scenes and subtitles more accurately. Experiments were conducted to test the performance of the proposed model by considering different Transformers, different input arrangements, subtitles with/without time stamp

annotations, and different maximum lengths with the proposed model. Results showed that BERT is the best-performed Transformer on the TVQA dataset, which improved the accuracy of video question answering by 4.23%, while ALBERT is the best-performed Transsformer on the Pororo dataset, which improved the accuracy by 6.73% compared with previous methods.

Moreover, further research regarding feature representations is conducted. The Transformers are deeper analyzed in video-QA tasks by comparing their accuracy of question answering under the same dataset. Results show that when different Transformers are adopted to answer questions in the same dataset, the accuracy of question answering has different values. The reason behind such differences is related to the different pre-training settings in different Transformers. It is suggested that the detailed situations of each task should be carefully considered before selecting Transformers for feature representation. This study revealed the mechanism of Transformer in feature representations and might offer some valuable suggestions to the selection of Transformers for similar tasks in the future.

For future work, there are mainly two different aspects to improve the performance of the proposed model. First, to broaden the application of video question answering system, the feature representations part can be improved. More related features, like action and relationship between characters, might be added into the input of the improved model, and certain methods will be used to give appropriate feature representations to them to make the improved method understand the video in a more comprehensive manner. Second, to confirm that the system really has a certain understanding of videos, it is necessary to make the improved model explainable to the reason of selecting certain candidate answers as the correct prediction, so that the improved system can become more credible and reliable.

Chapter 5

Multi-modal Humor Segment Prediction in Video

As mentioned in Section 1.7, this chapter introduces the dataset and the novel method for multi-modal humor segment prediction in video. A dataset is created based on the famous sitcom *the Big Bang Theory* to predict humor, which contains video frames, subtitles, and wave envelopes from 228 episodes in 10 seasons. Different from previous method that predicts humor based on sentences, the method proposed in this chapter uses a method based on sliding window to predict humor segments. A model that uses pose, face, and subtitle features is also built. Different combinations of pose and face features along with subtitle features are taken as inputs to know how they will perform in the prediction.

5.1 Introduction

Humor is the tendency to provoke laughter and provide amusement. It is an important medium to demonstrate our emotions and has become an essential tool in our daily life (Meyer, 2000). Humor can be used to draw people's attention and relieve stressful or embarrassing situations. By properly using humor, communications between people will become easier and smoother.

Most people are able to experience humor, which is usually induced by signals in visual (e.g., poses, gestures, and appearances), vocal (e.g., tones), and linguistic (e.g., puns) modalities (Hasan et al., 2019). They will be amused,



(a)

"Did you know it takes three sheep to make a sweater?" "I didn't know sheep could knit."

(c)



(b)

What do you call an alligator wearing a vest? An investigator.

(d)

Figure 5.1. Some examples of humor.

smile or laugh when the humor happens. Fig 5.1 are some examples that are humorous and will make people laugh.

It is an interesting task to let the computer understand humor. This task is also very necessary in the future because the computer needs to know us deeper in order to serve us well. By understanding humor better, the computer can have a deeper knowledge on how we think and how we behave under certain situation. It will then interact with us in a more comprehensive way and, ultimately, take our emotions into its decision strategies to respond to our various needs. For applications, humor prediction can be used in human-machine interactions to give better user experiences, it can also work on a robot to classify people's actions and better respond to related commands.

However, humor prediction is a very challenging task in both computer vision and natural language processing communities, because humor is induced from different modalities. A machine needs a deep understanding of not only signals in each modality but also their combinations to predict whether they cause people to laugh.

In recent years, some efforts have been made to predict humor both in sin-

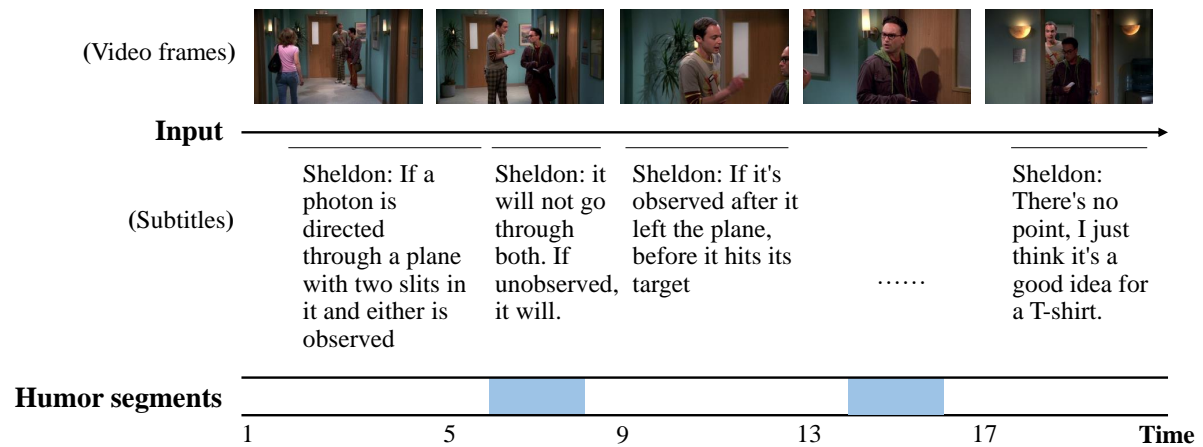


Figure 5.2. Overview of the task. This work is going to predict humor segments in the video.

gle modality and multiple modalities. Single modality methods usually model the linguistic modality to know whether the humor happens, while multiple modalities methods use the information from different modalities and combine them together for the final prediction. However, previous methods (Hasan et al., 2019; Weller & Seppi, 2019; Kayatani et al., 2021; Patro et al., 2021) are mainly driven by the linguistic modality. No matter what modality they are using, they mainly predict humor based on text (i.e. sentences, dialogs, etc.). The signals from different modalities often play a supplementary role. In the real world, however, the signals that cause humor may have different compositions, including some silent scenes with funny actions and facial expressions, which are often ignored in the tasks driven by the linguistic modality. To cover broader variations of humor, a new design for humor prediction task is presented.

With the motivation to know how well the artificial intelligence system understands humor, this work presents a new task for humor prediction. Unlike previous work that only predicts whether some sentences are humor or not, this work provides temporal segments in a video that are associated with humor as ground-truth and predicts such humor segments based on multi-modal input of video frames and subtitles. The concept of this task is demonstrated in Fig. 5.2.

To address this task, a method for humor prediction is introduced. This method uses sliding window to aggregate subtitles as well as pose and face features from the video frames in the window. The features are fed into the humor

prediction model to obtain the predictions of every sliding window. Then, these predictions are converted to temporal segments comparable with the ground-truth segments. This work distinguishes from previous methods (Kayatani et al., 2021; Patro et al., 2021) in that this work predicts humor based on temporal segments instead of a single sentence or a set of dialogues.

The contributions of this work are as follows:

1) A dataset is created based on *the Big Bang Theory* for video humor prediction. It contains video frames, subtitles, and wave envelopes. The wave envelope is binarized with thresholds to obtain humor segments. These humor segments are used as ground truths in the prediction.

2) A new task is proposed for humor prediction based on sliding windows. This task distinguishes from previous work in that it is going to find those temporal segments associated with humor instead of predicting humor based on sentences.

3) A method for multi-modal humor prediction is introduced. The visual modality (i.e. poses and faces of characters in video frames) and the linguistic modality (i.e. subtitles) are used as inputs and modelled in different ways.

4) Different combinations of input features are combined to show which feature combination is the best in humor prediction.

5.2 Prior Work

5.2.1 Humor Prediction Methods

Humor prediction methods have been viewed as a binary classification task. Generally, a humor prediction method takes the features obtained from text, image, and/or voice as inputs, and gives a prediction of whether there is humor as an output. In recent years, some work related to humor prediction has been proposed. For example, Bertero et al (Bertero & Fung, 2016a) propose a comparison between different machine learning methods to predict humor in dialogues; Kayatani et al (Kayatani et al., 2021) present a method to predict whether a subtitle causes laughter; Chen et al (Chen & Lee, 2017) build a presentation corpus containing humorous utterances based on TED talks and use convolu-

tional neural networks to automatically evaluating the humor; Chandrasekaran et al (Chandrasekaran et al., 2016) analyze the humor comic scenes and design computational models to predict them. Besides these methods, other attempts (Constantin, Redi, Zen, & Ionescu, 2019; Bertero & Fung, 2016b; Patro et al., 2021) have also been proposed to predict humor.

5.2.2 Humor Generation Methods

Humor generation is the task to generate verbal humor in the form of jokes and short humorous texts (Amin & Burghardt, 2020). The task might be beneficial to promote smooth interfaces between humans and machines based on natural language processing. Lots of work related to this task has been introduced. Zhang et al (H. Zhang, Liu, Lv, & Luo, 2020) explore how to generate a humorous sentence with the relevant knowledge by fusing the knowledge to end-to-end models; Weller et al (Weller, Fulda, & Seppi, 2020) propose a model to take normal text and convert it into humorous text to confirm how well the computer can generate humor; Binsted et al (Binsted & Ritchie, 1994) develop JAPE-1 to generates riddles from specific templates; Stock et al (Stock & Strapparava, 2005) presents HAHAcronym, whose main goal is to generalize humor acronyms. Besides these tasks, other strong attempts are also proposed (Sjöbergh & Araki, 2009; Dybala, Ptaszynski, Higuchi, Rzepka, & Araki, 2008; Winters, Nys, & De Schreye, 2019, 2018) and look into the task of humor generation.

5.3 The Proposed Datasets

A dataset based on the famous sitcom *the Big Bang Theory* is created for the humor prediction task. This dataset consists of video frames, subtitles, and wave envelopes. The detailed statistics of the dataset are shown in Table 5.1. An example for the dataset is shown in Fig. 5.3.

In the dataset, the video frames are re-sampled at 2 fps from the original videos. The subtitles are transcripts of utterances with the start and end timestamps. The wave envelope also comes from the original videos and is used as

Table 5.1

Statistics of the proposed dataset.

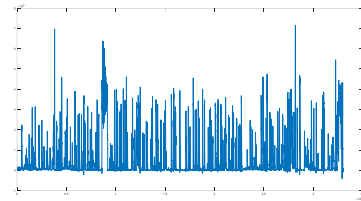
Number of Season		10
Number of Episode		228
Total Duration (hh:mm:ss)		76:33:50
Number of Humor Segments		31851
Humor Duration	Min (s)	0.042
	Avg (s)	2.254
	Max (s)	18.458
	Total (hh:mm:ss)	19:56:32
Number of Non-humor Segments		31963
Non-humor Duration	Min (s)	0.042
	Avg (s)	6.377
	Max (s)	76.792
	Total (hh:mm:ss)	56:37:18
Number of Subtitles	Total	74217
	Humor	33408
	Non-humor	40809



Video Frames

61-64
Receptionist: If you have to ask, maybe you shouldn't be here.
64-67
Sheldon: I think this is the place.
68-69
Receptionist: Fill these out. Thank you.

Subtitle



Wave Envelope

Figure 5.3. The proposed dataset for humor prediction, which contains video frames, subtitles, and wave envelopes.

the main source to find ground-truth humor.

One main problem in creating the dataset is how to find humor segments. In this dataset, by following previous work (Kayatani et al., 2021), the humor segments can be found based on laughter. The main motivation for doing this is that this selection is relatively objective because the sitcom usually has a laugh track to record the laughter made by the audience or the canned laughter added by the producer. In the video, the audio tracks mix with the laugh track and the music track. Thus, this audio track can be used to extract laughter and find humor segments. Note that this dataset makes no difference between the

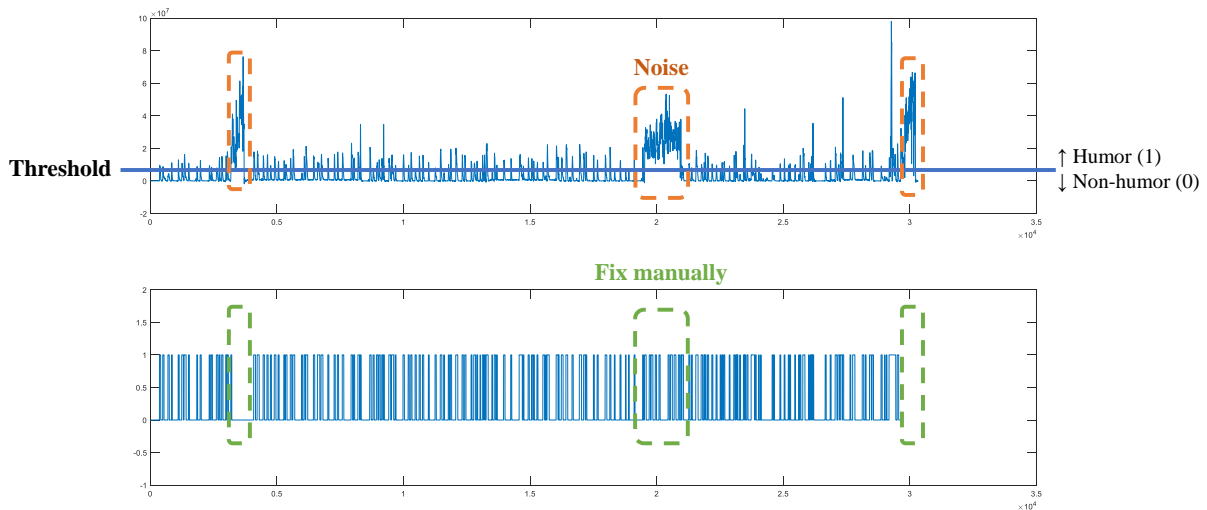


Figure 5.4. The ground-truth humor segments can be obtained by setting up threshold and binarizing the wave envelope. The boxes in orange are wave envelopes with noise, and the boxes in green are the fixed humor segments.

audience’s laughter and the canned laughter as it is difficult to distinguish them by only referring to the information in audio tracks.

For the wave envelope, the left and right channels of the audio track are first subtracted to cancel out the characters’ speech. Then, the wave envelope is down-sampled to 24fps. Next, low-pass filtering is applied to push out the laughter. Thresholds are set up and binarize the envelope, giving 1 for humor and 0 for non-humor to obtain the ground-truth humor segments, which is shown in Fig. 5.4. As the envelope is easily influenced by noise, including sound effects and jingles, all the extracted segments are checked manually to fix the false annotations.

5.4 The Proposed Model

The main task in this research is to localize the humor segments based on video frames and subtitles. For this task, humor segment prediction task is converted into humor/non-humor prediction over sliding windows to model the dependency among them. To predict humor over sliding windows, three flows are used to model the information in the video frames and subtitles respectively.

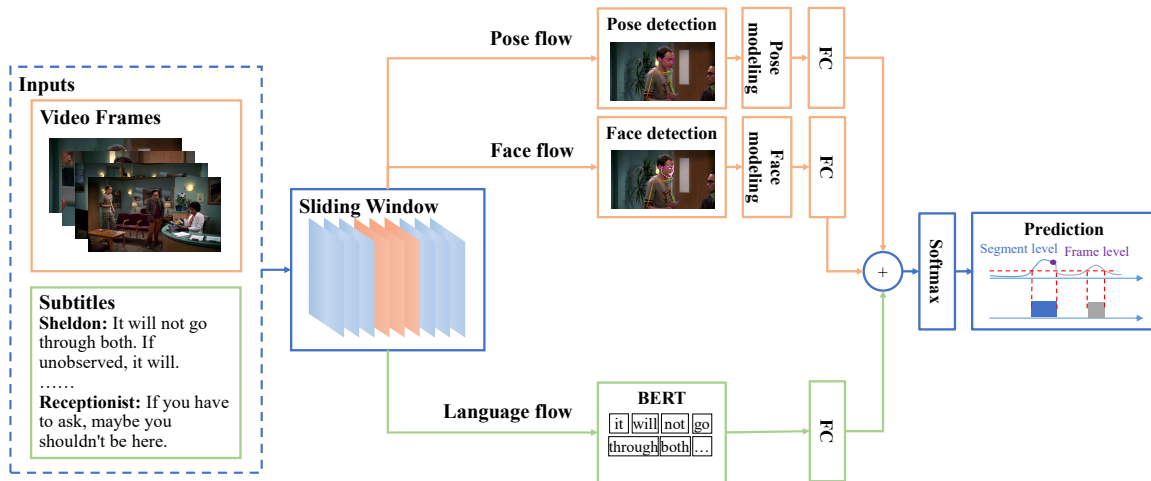


Figure 5.5. The proposed model for multi-modal humor prediction.

The video frames are modeled to form the pose flow and the face flow, and subtitles are modeled to form the language flow. The prediction scores of different flows are summed later to obtain the per-window predictions. Then, per-window predictions are converted to predicted segments based on the time stamps of each sliding window. The overview of the proposed method is shown in Fig. 5.5.

5.4.1 Sliding Window

A humor usually has two parts that complement each other: the preparation part and the triggering part. In the preparation part, people give some signals that are related to humor, but these signals will not cause laughter. In the triggering part, the given signals finally make the audience laugh. This work assumes that humor is triggered in a certain period and uses a sliding window with fix-length T to simulate it. Then, this window moves by a fixed-length time step t in the timeline ($t \leq T$) to divide the long video.

For the i -th window w_i , the video frames $V_i = \{v_{ij} | j = 1, \dots, J\}$ and subtitles $S_i = \{s_{ik} | k = 1, \dots, K_i\}$ within it are aggregated as the input of the model, where J and K_i are the numbers of frames and subtitles in w_i (K_i can vary for different windows), respectively. Note that the subtitle that ends inside the window will

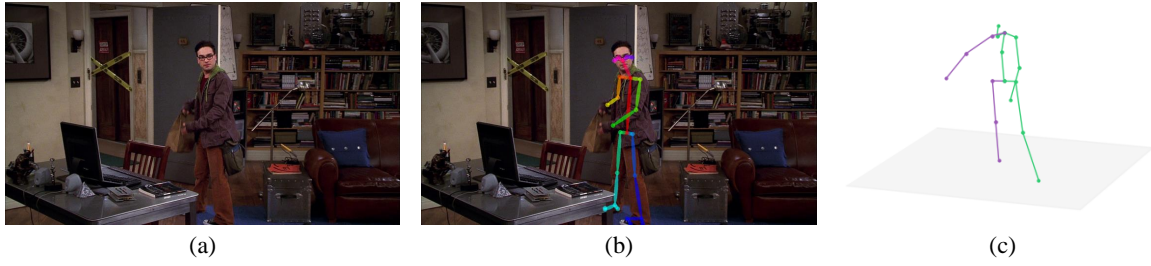


Figure 5.6. Examples of original video frame and pose detection. (a) Video frames. (b) 2D pose. (c) 3D pose

be included, while the subtitle that starts inside, but ends outside the window will not be included. The neural network-based model make humor/non-humor prediction h_i for the i -th sliding window w_i .

In previous work, BERT models language features well, yet it also has a good performance in sentence-level humor predictions (Kayatani et al., 2021; Patro et al., 2021). Thus, the proposed method model the dependency among all subtitle s_{ik} in S_i with BERT. For the visual modality, two flows are used to represent poses and facial expressions of the characters in w_i respectively, since humor is sometimes induced by funny poses and facial expressions.

5.4.2 Pose Flow

Some funny actions make people laugh. Poses of characters in the video frame demonstrate what actions they are doing and their features are crucial for humor prediction. To process pose features, the model has a pose flow. Two different kinds of pose features are compared to find the better-performing one:

- 1) Use OpenPose (Cao, Hidalgo, Simon, Wei, & Sheikh, 2019) to detect joint positions in the video frames in V_i . For each person in each video frame, a $3M$ -D vector containing the 2-D coordinates of each joint with the confidence score given by OpenPose is obtained, where $M = 25$.

- 2) Convert the 2-D joint coordinates to 3-D coordinates with OpenPose 3D baseline (Martinez, Hossain, Romero, & Little, 2017) pre-trained on Human 3.6M dataset (Ionescu, Papava, Olaru, & Sminchisescu, 2013), which maps

$M = 25$ into $M' = 32$ with 17 movable joints and 15 fixed joints to fit the Human 3.6M model. These 17 movable joints are found to obtain a vector containing all of their coordinates in 3D space.

Examples of 2D and 3D poses are shown in Fig. 5.6 respectively.

The confidence score c_m^P for each joint ($m = 1, \dots, M$) may somehow represent the importance of the corresponding person in the video scene. Based on this idea, the method calculates the average confidence score \bar{c}^P for each person by:

$$\bar{c}^P = \frac{1}{M} \sum_{m=1}^M c_m^P \quad (5.1)$$

Then, the characters in the video scene are ranked based on \bar{c}^P and select top-3 characters for both 2-D and 3-D poses. Note that the 2-D confidence information ($M = 25$) is still used for 3-D poses because 3-D poses are created based on 2-D poses.

Let x^P denote the vector of either kind of pose features for character, x^P is fed into FC layers and max-pool them to obtain a 128-D pose vector p_{ij} for each frame. p_{ij} are concatenated and fed into an LSTM layer. The hidden state corresponding to the last frame (i.e., v_{iJ}) is then fed into an FC layer to get the prediction score $e_i^P \in \mathbb{R}^2$ for the pose flow.

5.4.3 Face Flow

Some exaggerated facial expressions also make people laugh. It is also essential to model them to predict the laughter. The face expressions are modeled in a similar way as the pose flow. Two kinds of facial features are used to find the better performing one:

1) Use a variant of OpenPose to detect facial landmark in the video frames in V_i . For each person in V_i , a $3N$ -D vector containing the 2-D coordinates of face landmark with the confidence score c_n^F is given by OpenPose, where $N = 70$.

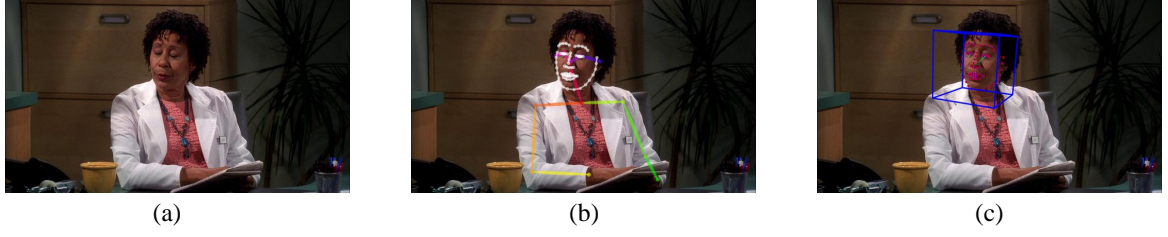


Figure 5.7. Examples of original video frame and face detection. (a) Video frames. (b) Face landmark. (c) Action unit.

The average confidence score \bar{c}^F is calculated for each person:

$$\bar{c}^F = \frac{1}{N} \sum_{n=1}^N c_n^F \quad (5.2)$$

2) Use Openface (H. Liu et al., 2018) to extract a N' D vector of action unit from each character in a video frame and an average confidence score \bar{c}^F , where $N' = 35$.

Examples of face landmarks and action units are shown in Fig. 5.7 respectively.

No matter which feature is used, the top 3 characters with the maximum \bar{c}^F score are selected. Their landmarks or action units, x^F are fed into FC layers and maxpool them to obtain a 128D face vector f_{ij} for each frame. Then, f_{ij} in different frames are concatenated and fed into an LSTM layer. The prediction score for the face flow is denoted by e_i^F .

5.4.4 Language Flow

For the language flow, the subtitles in the video will be modeled as they contain the transcription of what the characters say and are the primary source to makes people laugh. It can be found that BERT (Devlin et al., 2018) has been widely applied in many similar tasks for language feature representations, and it shows outstanding results (Devlin et al., 2018; Ma, Xu, Wang, Nallapati, & Xiang, 2019; Z. Zhang et al., 2020). Thus, BERT is used in the proposed method to model the subtitles. All the subtitles in S_i are concatenated. Special tokens

including [CLS] are added to convey the subtitles into BERT. Only these special tokens are passed to BERT if there are no subtitles in the sliding window. The output corresponding to [CLS] is then fed into an FC layer to get the prediction score $e_i^S \in \mathbb{R}^2$.

5.4.5 Prediction and Training

In this model, late fusion is adopted for the final prediction. All the prediction scores are summed together to get e_i and e_i will go through a softmax layer to obtain the final score E_i :

$$e_i = e_i^P + e_i^F + e_i^S \quad (5.3)$$

$$E_i = \text{softmax}(e^i) \quad (5.4)$$

The humor label, h_i , for sliding window w_i , is predicted by selecting the binary index with the higher prediction score in E_i , that is:

$$h_i = \max(E_i) \quad (5.5)$$

For training, considering that laughter happens after a triggering part, window w_i is set as being associated with humor if the end time of the window falls within the ground-truth humor segments, and otherwise it is considered as non-humor.

5.4.6 Converting Frame Predictions to Temporal Segments

For a single sliding window w_i , a binarized prediction of humor label h_i is given to show whether the end of this frame causes humor or not. This label is called the frame-level prediction. The segment-level prediction showing how the predicted humor segment is aligned is obtained by converting the frame-level predictions to predicted humor segments. As t is the shift amount between consecutive windows, if w_i is predicted as humor, the following t second will be considered as a humor segment. Consecutive segments are merged together to

Table 5.2

The training parameters in the experiment.

Parameters	Setting
Learning Rate	2×10^{-5}
Number of Epoch	3
Training Batch Size	32
Inferring Batch Size	16
Max Number of Tokens	128
Optimizer	Adam
Weight Decay	1×10^{-5}

form a humor segment, which is the segment-level prediction of humor.

5.5 Experimental Results

The proposed method is implemented based on Python 3 and Pytorch. The training parameters in the experiments are shown in Table 5.2. In the experiments, 80% of data is split as the training set, 10% is split as the validation set and 10% is split as the test set. For the subtitle input, bert-base-uncased model with 12 layers, 768 hidden sizes, 12 self-attentions and 110 million parameters is used for feature representation. The model makes no distinction between upper case and lower case tokens. Cross-entropy loss is applied for training.

In the proposed dataset, the average length of humor is 2.25 s and the average length of non-humor is 6.38 s. To let the sliding window cover most humor segments in the videos, the length of sliding window is set to $T = 8$ s and the shift is set to $t = 2$ s.

5.5.1 Quantitative Results

The frame-level and segment-level prediction results of the proposed method with different ablation studies are shown in Table 5.3 and Table 5.4 respectively. For a comprehensive evaluation, accuracy (Acc), precision (Pre), recall (Rec) and F1 are taken as metrics for frame-level prediction and precision (Pre), recall (Rec) and F1 under different IoU thresholds are taken as metrics for segment-

Table 5.3

Frame level results on the Big Bang Theory test set (in %)

Pose	Input		Acc	Pre	Rec	F1
	Face	Subtitle				
	All Positive		32.00	32.00	100.00	48.49
	All Negative		68.00	0.00	0.00	0.00
2D	—	—	68.84	66.27	5.36	9.92
3D	—	—	68.57	70.95	3.04	5.83
—	landmark	—	68.93	66.31	5.93	10.89
—	Action Unit	—	67.98	33.33	0.02	0.05
—	—	BERT	70.23	55.93	32.97	41.48
2D	landmark	—	68.97	63.71	7.06	12.71
2D	Action Unit	—	68.62	59.62	6.00	10.91
3D	landmark	—	68.86	65.15	5.81	10.67
3D	Action Unit	—	68.56	70.33	3.06	5.87
2D	—	BERT	70.75	57.23	34.09	42.73
3D	—	BERT	70.94	57.53	35.10	43.60
—	Landmark	BERT	71.01	57.38	36.56	44.66
—	Action Unit	BERT	70.16	59.76	20.65	34.69
2D	Landmark	BERT	70.94	57.48	33.41	43.82
2D	Action Unit	BERT	70.51	59.56	24.45	34.67
3D	Landmark	BERT	71.12	57.87	35.89	44.30
3D	Action Unit	BERT	70.33	56.08	33.66	42.07

level prediction. The results on two naive baselines are also shown, where all positive and all negative labels are used as the predicted humor. The subtitle baseline, which uses subtitles within the sliding window as language flow input and the other (i.e., pose and face) flows are removed, is also employed.

Besides those baselines, Kayatani et al’s method is also taken as the baseline as well, though their task mainly predicts the humor based on sentence. For a comparable result, their sentence-level prediction is converted into frame-level predictions by making use of the time stamp of each sentence. As the average length of humor is about 2 s, for a sentence predicted as humor, all the frames within the range from the end of the sentence to 2 second later to the end of that sentence are set as its humor segment. The frame-level and segment-level results are shown in Tables 5.5 and 5.6, respectively. The word *Char* in the

Table 5.4

Segment level results on the Big Bang Theory test set (in %)

Input			IoU=0.25			IoU=0.50			IoU=0.75		
Pose	Face	Subtitle	Pre	Rec	F1	Pre	Rec	F1	Pre	Rec	F1
2D	—	—	63.00	6.88	12.40	42.49	4.64	8.36	15.75	1.72	3.10
3D	—	—	64.25	4.60	8.58	39.66	2.84	5.30	8.94	0.64	1.19
—	Landmark	—	64.00	7.04	12.68	41.45	4.56	8.21	15.27	1.68	3.02
—	Action Unit	—	33.33	0.04	0.08	0.00	0.00	0.00	0.00	0.00	0.00
—	—	BERT	75.49	35.23	48.04	52.61	24.58	33.51	20.87	9.75	13.29
2D	Landmark	—	60.28	8.56	14.99	43.10	6.12	10.71	19.44	2.76	4.83
2D	Action Unit	—	56.06	8.88	15.33	36.36	5.76	9.94	16.41	2.60	4.49
3D	Landmark	—	63.33	6.84	12.34	40.37	4.36	7.86	15.93	1.72	3.10
3D	Action Unit	—	62.98	4.56	8.50	39.23	2.84	5.30	9.39	0.68	1.27
2D	—	BERT	74.65	36.27	48.82	51.97	25.30	34.03	21.18	10.31	13.84
3D	—	BERT	76.07	37.00	49.78	54.44	26.48	35.63	22.29	10.84	14.59
—	Landmark	BERT	76.00	37.98	50.65	52.95	26.54	35.36	20.41	10.23	13.63
—	Action Unit	BERT	76.77	24.71	37.39	55.21	17.79	26.90	22.08	7.11	10.76
2D	Landmark	BERT	76.48	38.34	51.08	52.07	26.14	34.81	21.02	10.55	14.05
2D	Action Unit	BERT	76.46	28.83	41.87	55.34	20.90	30.35	23.39	8.83	12.82
3D	Landmark	BERT	75.35	38.38	50.86	52.85	27.02	35.76	21.74	11.11	14.71
3D	Action Unit	BERT	75.47	35.19	48.00	51.46	24.02	32.75	20.03	9.35	12.75

Table 5.5

Frame level results of Kayatani et al's method and the proposed method (in %)

Method	Input			Acc	Pre	Rec	F1
Kayatani et al	—	—	BERT	62.94	44.10	59.04	50.49
	—	Action Unit	BERT	65.20	46.14	52.13	48.95
	Char	Action Unit	BERT	66.71	47.40	39.88	43.40
Proposed	—	—	BERT	70.23	55.93	32.97	41.48
	—	Landmark	BERT	71.01	57.38	36.56	44.66
	3D	Landmark	BERT	71.12	57.87	35.89	44.30

tables denotes character features in Kayatani et al's method.

Table 5.3 shows the frame-level results with ablation studies. In the table, the ablation accuracy of different input modalities in the proposed method is more than 60%. When only pose and/or face features are adopted, the recall

Table 5.6

Segment level results of Kayatani et al's method and the proposed method (in %)

Method	Input	IoU=0.25			IoU=0.50			IoU=0.75		
		Pre	Rec	F1	Pre	Rec	F1	Pre	Rec	F1
Kayatani et al	BERT	66.18	42.02	51.41	36.34	23.07	28.22	10.14	6.44	7.87
	Action Unit + BERT	68.51	39.06	49.76	40.39	23.03	29.34	11.01	6.28	8.00
	Char + Action Unit + BERT	69.87	33.11	44.93	45.65	21.63	29.35	14.68	6.96	9.44
Proposed	BERT	75.49	35.23	48.04	52.61	24.58	33.51	20.87	9.75	13.29
	Landmark + BERT	76.00	37.98	50.65	52.95	26.54	35.36	20.41	10.23	13.63
	3D + Landmark + BERT	75.35	38.38	50.86	52.85	27.02	35.76	21.74	11.11	14.71

and F1 are low. When the language features are also taken into the input, the accuracy, recall, and F1 get improved. Moreover, the best-performed modalities under the proposed metrics are all with language features. The reason might be that most humors in the dataset are triggered from the linguistic modality, while funny poses and facial expressions are fewer compared with those humorous words. However, funny poses and facial expressions do have contributions in humor prediction, as the precision with visual modality only is more than 65%. The one with 3D pose, face landmark and subtitle inputs has better accuracy, while the one with face landmark and subtitle inputs has better recall and F1 than other input modalities.

Different IoU thresholds are set up between the predicted segments and ground truth segments to evaluate the proposed method in segment level. The segment-level evaluation is shown in Table 5.4. The results show that the precision, recall, and F1 scores in segment level drop when the IoU threshold increases. Also, language features have a large contribution to the improvement in the prediction: When the inputs contain subtitle features, the recall and precision are much better than those with only visual features (pose feature and/or face feature). Among the results, the one with 3D pose, face landmark and subtitle inputs has the best recall when $\text{IoU} = 0.25$, $\text{IoU} = 0.50$ and $\text{IoU} = 0.75$. It also has the best F1 values over other inputs when $\text{IoU} = 0.50$ and $\text{IoU} = 0.75$, being the best-performing inputs in segment-level evaluation.

Next, the results between the proposed method and Kayatani et al's method (Kayatani et al., 2021) are analyzed. The test accuracy and precision of the pro-

posed method in the frame level with 3D pose, face landmark, and subtitle features are 4.41% and 10.47% higher than Kayatani et al's method using character, action unit, and subtitle features and predicts humor in sentence level. Even if the recall of the proposed method is 3.99% lower, the F1 score is still 0.90% higher than their method. This means that the proposed sliding-window-based method has a more accurate prediction and achieves a better balance between precision and recall than the method in sentence level. In segment level, the best recall and F1 are 3.64% and 0.55% lower than the best value in Kayatani et al's method when $\text{IoU} = 0.25$. When $\text{IoU} = 0.50$, the best precision, recall and F1 of the proposed method are 7.30%, 3.95% and 6.41% higher than Kayatani et al. When $\text{IoU} = 0.75$, the best precision, recall and F1 of the proposed method are 7.06%, 4.15% and 5.27% higher. This means that humor prediction method in sentence level catches humor roughly, but its alignments of the predicted humor segment are not good. The proposed sliding-window-based method may miss some humor, but its alignments of predicted humor segments are better than the method in sentence level.

5.5.2 Qualitative Results

Some examples of the prediction using 3D pose, face landmark, and subtitles are shown along with the predictions by Kayatani et al's method (Kayatani et al., 2021) using character, face, and subtitle features in Fig. 5.8. The green bars in the timeline indicate the predicted humor segments with the proposed method, the orange bars indicate the predicted humor segments using Kayatani et al's method and the blue bars indicate the ground truth humor segments. The IoU of the proposed method and Kayatani et al's method with ground truth humor segments are calculated and marked inside the bars respectively. From the figures, the predicted humor segments using the proposed method have a higher IoU value than Kayatani et al's method in (a), (b) and (c), but has a lower IoU value in (d). Now, each case will be discussed in detail.

In (a), the humor is mainly triggered by funny actions (one of the actors is lifting his arms in front of his chest), the proposed method captures these actions and finds the corresponding segments. Kayatani et al's method also

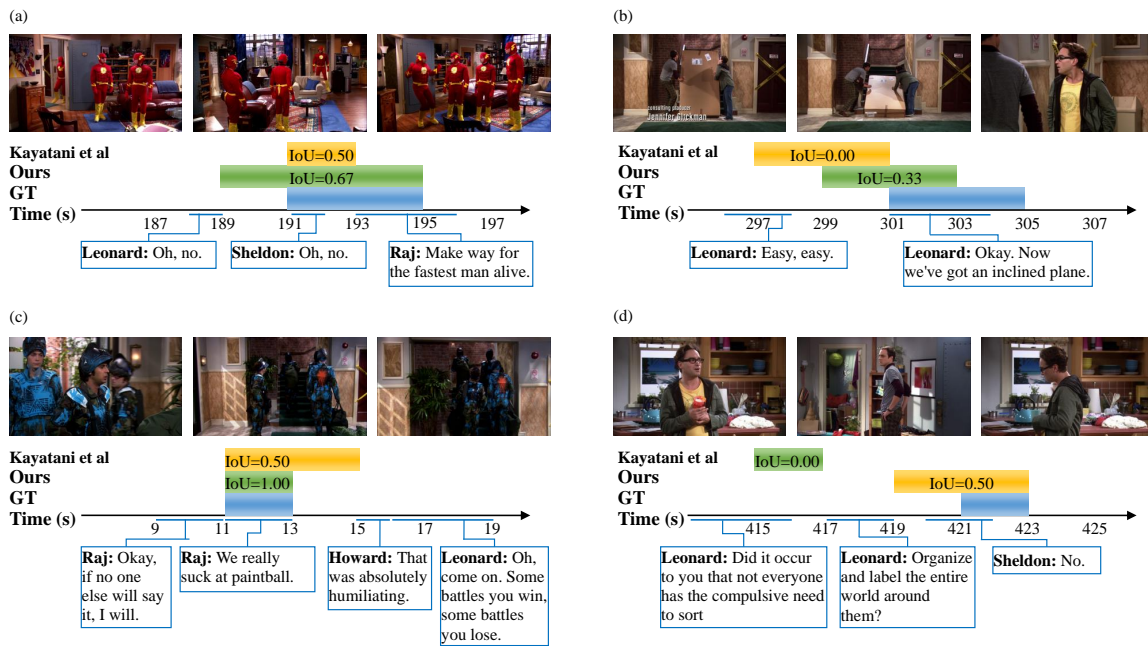


Figure 5.8. Examples of humor predictions using the proposed method and Kayatani et al's method.

catches humor, but its IoU is lower than the proposed method. In (b), the humor is mainly caused by two persons lifting a wooden board together. The proposed method predicts the segment with some overlap with ground truth, but it starts earlier and ends earlier, while Kayatani et al's method fails to predict the humor segment correctly. In (c), even if the actors are wearing special costumes, the segment that causes humor is limited. The proposed method captures and predicts the humor segment correctly, while Kayatani et al's method thinks the humor should end later. In (d), the proposed method fails to capture the humor segment, while Kayatani et al's method catches it. The reason might be that the humor is triggered because *Leonard* speaks something to *Sheldon*. The proposed method does not take the relationship between different characters into consideration, it has no information about which person is speaking to which person and thus gives a bad prediction.

Table 5.7

Training time on the Big Bang Theory dataset per epoch (in mm:ss)

Input	Training time
Pose	1:06
Pose+Face	2:01
Pose+Subtitle	20:08
Face	1:07
Face+Subtitle	20:46
Subtitle	19:45
Pose+Face+Subtitle	23:07

5.6 Discussion

5.6.1 Training Time

It is important to know how long does it take for training when using the proposed method, so the time consumed in training every epoch is reported. The experiments are performed on a computer with Intel Core i7-8700K CPU, 32G RAM and a Nvidia Titan RTX GPU. The training time per epoch using different inputs is shown in Table 5.7.

From the table, when subtitle is not taken into the input of the model, the training per epoch can be finished in no more than 3 min. However, when subtitle is taken into the input, the training per epoch costs more than 19 min to finish. This is because pre-trained BERT needs more memory to represent language features compared with LSTM networks, especially when it meets long sentences. This process makes the training both time-consuming and memory-consuming.

5.6.2 Training with All the Joints in 3D Pose

In Section 5.4, Openpose 3D baseline (Martinez et al., 2017) is used to convert the 25-joint 2D pose into 32-joint 3D pose, which contains 17 movable joints and 15 fixed joints. It is interesting to know how the prediction will be influenced by using all 3D joints. Thus, the coordinates of all the 32-D joints are taken for humor prediction, and the results of frame-level prediction and

Table 5.8

Frame level results on the Big Bang Theory test set with all joints in 3D pose (in %)

Pose	Input		Acc	Pre	Rec	F1
	Face	Subtitle				
3D-17 joints	—	—	68.57	70.95	3.04	5.83
3D-32 joints	—	—	68.55	70.00	3.01	5.78
3D-17 joints	landmark	—	68.86	65.15	5.81	10.67
3D-17 joints	Action Unit	—	68.56	70.33	3.06	5.87
3D-32 joints	Landmark	—	68.52	65.71	6.00	11.00
3D-32 joints	Action Unit	—	68.56	70.39	3.01	5.78
3D-17 joints	—	BERT	70.94	57.53	35.10	43.60
3D-32 joints	—	BERT	70.23	55.97	32.75	41.32
3D-17 joints	Landmark	BERT	71.12	57.87	35.89	44.30
3D-17 joints	Action Unit	BERT	70.33	56.08	33.66	42.07
3D-32 joints	Landmark	BERT	71.11	58.14	34.76	43.51
3D-32 joints	Action Unit	BERT	70.11	55.57	32.94	40.36

Table 5.9

Segment level results on the Big Bang Theory test set with all joints in 3D pose (in %)

Pose	Input		IoU=0.25			IoU=0.50			IoU=0.75		
	Face	Subtitle	Pre	Rec	F1	Pre	Rec	F1	Pre	Rec	F1
3D-17 joints	—	—	64.25	4.60	8.58	39.66	2.84	5.30	8.94	0.64	1.19
3D-32 joints	—	—	62.78	4.52	8.43	39.44	2.84	5.29	8.89	0.64	1.19
3D-17 joints	Landmark	—	63.33	6.84	12.34	40.37	4.36	7.86	15.93	1.72	3.10
3D-17 joints	Action Unit	—	62.98	4.56	8.50	39.23	2.84	5.30	9.39	0.68	1.27
3D-32 joints	Landmark	—	62.95	7.00	12.59	41.73	4.64	8.35	18.71	2.08	3.74
3D-32 joints	Action Unit	—	63.13	4.52	8.43	39.66	2.84	5.30	8.38	0.60	1.12
3D-17 joints	—	BERT	76.07	37.00	49.78	54.44	26.48	35.63	22.29	10.84	14.59
3D-32 joints	—	BERT	75.13	34.67	47.44	51.82	23.94	32.75	19.75	9.11	12.47
3D-17 joints	Landmark	BERT	75.35	38.38	50.86	52.85	27.02	35.76	21.74	11.11	14.71
3D-17 joints	Action Unit	BERT	75.47	35.19	48.00	51.46	24.02	32.75	20.03	9.35	12.75
3D-32 joints	Landmark	BERT	76.04	37.43	50.16	52.51	25.90	34.69	20.83	10.27	13.76
3D-32 joints	Action Unit	BERT	74.81	34.67	47.38	51.64	23.98	32.75	20.05	9.31	12.72

segment-level prediction are shown in Table 5.8 and 5.9, respectively.

From the results, it can be seen that when only 17 movable joints are used, the prediction has generally better results in both frame level and segment level. This implies that the action itself can be demonstrated by the movable joints

Table 5.10

Frame-level results on the Big Bang Theory test set under different lengths (in %)

Input	Length	Acc	Pre	Rec	F1
Pose+Face+Subtitle	4s	68.82	52.65	24.16	33.73
	8s	71.12	57.87	35.89	44.30
	12s	69.71	56.52	23.78	33.47
	16s	69.02	56.20	15.36	24.13

directly. However, when only the pose and face features are taken as inputs, the input with 32 joints has better results than 17 joints. This tells that the facial expressions might have some connections with the fixed joints for a better humor prediction.

5.6.3 Training with Different Lengths of Sliding Window

In the proposed method, the length of sliding window plays a vital role in determining how much information can be conveyed into the neural network. It is also important to know whether the length of sliding window has some influence on the performance of humor prediction, and how. Hence, different values are selected as the length of sliding window and the performance of the proposed method under them is evaluated. 3D pose and face landmark are used as inputs of pose flow and face flow, respectively, and subtitle features are also modeled in the language flow. The lengths of sliding window, T , is set to 4 s, 8 s, 12 s, and 16 s, respectively, while the shift length of sliding window, t , is kept to 2 s. The performances of humor prediction in frame-level and segment-level under different lengths of sliding window are shown in Table 5.10 and Table 5.11, respectively.

From Tables 5.10 and 5.11, when the sliding window gets longer, both frame-level and segment-level results increase at first, and decrease later. Both of them obtain the highest value when the length of sliding window $T=8$ s. This tendency shows that the information in the short sliding window is not enough to predict humor, while the information in the long sliding window may lead to a drop in the performance of the proposed method. The reason might be

Table 5.11

Segment-level results on the Big Bang Theory test set under different lengths (in %)

Input	Length	IoU=0.25			IoU=0.50			IoU=0.75		
		Pre	Rec	F1	Pre	Rec	F1	Pre	Recall	F1
Pose+ Face+ Subtitle	4s	62.50	30.14	40.67	45.70	22.06	29.76	15.95	7.70	10.38
	8s	75.35	38.38	50.86	52.85	27.02	35.76	21.74	11.11	14.71
	12s	71.53	23.57	35.45	46.48	15.36	23.09	16.14	5.33	8.02
	16s	68.11	14.60	24.05	45.69	9.82	16.16	16.67	3.58	5.90

that the humor is often triggered in a relatively short period. When the sliding window is long, some information that is not related to humor will be conveyed into the network, making the performance drop. Also, the proposed dataset contains lots of humor that are mainly based on the language modality, which needs several rounds of dialogue to be triggered. When the sliding window is too short, the information in the sliding window is not enough to be connected with laughter, and thus, difficult to predict whether it is humor or not. Hence, the sliding window should be in an appropriate size for a better prediction.

5.7 Summary

This chapter proposed a task for deep video understanding by carrying out humor prediction task on sitcom videos. A dataset was introduced to predict humor based on the famous sitcom *the Big Bang Theory*. 228 episodes from 10 seasons were selected and the video frames, the subtitles and the wave envelopes were used as sources for humor prediction. This chapter also presented a framework to predict video humor in multiple modalities. The method predicts humor segments, which distinguishes from previous methods that predict humor based on a single sentence. For this goal, a method was also taken to use features from subtitles along with different kinds of pose and face features in the videos to predict humor. The proposed method used BERT to model the subtitles and set up LSTM networks to model pose and facial expression features.

Experiments were carried by selecting different input feature representations, setting different input flows and changing the length of sliding window. The

results showed that the proposed method outperformed the previous method that predicts humor based on sentences, and gave better humor predictions in both frame level and segment level. The one that uses 3D pose, face landmark and subtitle inputs performed the best in both frame-level and segment-level predictions. This means that the proposed method gave a better representation for visual and language features and showed a deeper understanding of videos.

However, the proposed method still faces some limitations. First, it mainly focuses on modelling the people themselves in the video frames. Once the humor is triggered from other objects (like a funny costume or property) or specific relationships between characters, this method cannot model them, and would be hard to predict whether humor happens. Solutions might be to model the objects and the relationship between people in order to give a better feature description. Second, the dataset used in this method is mainly based on a sitcom, whose aim is to make people laugh. However, in the real world, people's emotions might be demonstrated in various ways, the prediction will be difficult if other emotions are taken into consideration. It is important to broaden the source of the dataset, import outside knowledge bases, and migrate the prediction to other emotions. Third, this method only gives a binarized classification about whether humor happens or not, but in the real world, different humor might have different strengths and categories, and people might respond to them differently. Research that may go one step further is to classify different kinds of humor by their type and people's response, so that the computer can have a deeper understanding not only to the video itself, but also to the content of humor.

Chapter 6

Conclusion and Outlook

As described in Section 1.7, this is the final chapter of the thesis that presents a summary of the main findings of the selected research project and provides some suggestions regarding future studies.

6.1 Conclusion

This thesis, with the purpose of promoting deeper understanding of videos, mainly focused on the study of feature representations in visual and language tasks. The study is conducted based on two different tasks with close relationships to both visual and language inputs, and looked into feature representations in three different aspects. In this section, the main findings in this study are concluded.

First, this thesis selected different features to describe visual and language elements in two tasks respectively. In video question answering task, visual concept features that are in the form of words or phrases were selected to reflect the visual information, while subtitles that reflect the people’s dialogues were selected to reflect language information. In humor prediction tasks, pose features and facial features that are in the form of vectors were selected to reflect the visual information, while subtitles were selected to reflect language information. Experiments showed that the selected features covered most of the information in the visual and language inputs and helped understand the video deeper.

Second, this thesis focused on finding better ways to give feature representa-

tions in two different tasks by considering their different characteristics. In the video question answering task, BERT and other three Transformers were taken to give representations to deep semantics in the visual and language inputs. In the humor prediction task, BERT was taken to give representations to subtitles, while LSTM was taken to give representations to visual features. Experiments were carried in both tasks to compare the performance of the proposed methods with previous methods. Results showed that both methods outperformed previous methods, and promoted a deeper understanding of video contents.

Based on the experimental results, both the selected features and the proposed methods regarding feature representations are appropriate for the selected tasks. They helped improve the performance in video question answering and humor prediction tasks compared with previous studies. This study finally reached the goal of giving a deeper understanding of videos.

6.2 Outlook

Visual and language task processes the image and the language simultaneously and has become a relatively new branch in artificial intelligence. It not only builds a bridge between the image and the language, but also sets up another way for deeper information understanding. With the rapid development of artificial intelligence, both visual and language will be essential for the computer to communicate with people better. An increasing number of applications that involve visual and language will also be developed, giving a large number of possibilities to open up this field and meet various demands in different aspects. However, the way towards this goal has both opportunities and challenges. In this section, some suggestions are given regarding future studies to make the performance of visual and language tasks in the future further improved:

- 1) Since the goal for each task is different, it is necessary to weigh pros and cons and select proper features that are suitable for the task from visual and language inputs. It is also important to select proper pre-trained model to represent these features by considering the goal of the task, the style of the feature, and the characteristics of each method.

2) Compared with features obtained from language elements, features obtained from images are still not accurate enough because image features are more difficult for computers to process. To improve image feature retrieving, it is essential to improve the object detection mechanism to fully find the features that are needed. Next, a proper description of the features in images is also required. By improving object detection and feature description, the information in the image can be conveyed to the computer in a more comprehensive way.

3) In some tasks related to visual and language, the system makes predictions based on a sequence of images that has certain connections in the temporal domain (like the video frames used in the selected tasks) instead of a single image. To handle this task, the system not only needs to know the evolution between the image sequences, but also has to mine the image sequence to find some unseen information (like the relationship between people) to make the prediction become more accurate.

4) To handle visual and language tasks, a deep understanding of both visual and language elements is required. However, the information provided in the image and the language may not be enough in some cases, making it necessary to explore potential feature representations for a correct prediction. For this point, knowledge representation or object relations based shall be explored on visual and language inputs, so that the system can be more intelligent.

5) The process of prediction-making in visual and language tasks is still in a black box, making it doubtful to know whether the task really understands the visual and language contents or not. To confirm this point and judge whether visual and language task gives prediction based on certain features or by random guess, it has become a tendency to make the visual and language task able to explain the reason of giving certain predictions, so that it can become more credible and reliable.

6) The features in visual and language tasks are firstly represented using high-dimensional vectors. Then, they will be conveyed to the neural networks and finally obtain the prediction for this task. This whole process is both time-consuming and memory-consuming, and might give a heavy burden to the local memory of an embedded system (e.g. robots, unmanned aircraft, human-

machine interaction equipment). Certain compression and pruning are required in the network in order to reduce the parameters and achieve a balance between the size and the performance.

7) Visual and language tasks might be applied to solve problems in different fields, like journalism, geology, medical, and pharmacy. There will be both opportunities and challenges to combine artificial intelligence techniques with different professional skills and develop a system that is suitable for them. Close collaboration is required between those people with different kinds of professional skills in various aspects.

In all, as an important branch in artificial intelligence, visual and language task will have many potential applications in the years to come. By looking into visual and language tasks, we can not only explore how a computer processes information, but also share the convenience it brings to create a better world for people.

References

- Abu-El-Haija, S., Kothari, N., Lee, J., Natsev, P., Toderici, G., Varadarajan, B., & Vijayanarasimhan, S. (2016). Youtube-8m: A large-scale video classification benchmark. *arXiv preprint arXiv:1609.08675*.
- Agrawal, A., Batra, D., Parikh, D., & Kembhavi, A. (2018). Don't just assume; look and answer: Overcoming priors for visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 4971–4980).
- Amin, M., & Burghardt, M. (2020). A survey on approaches to computational humor generation. In *Proceedings of the 4th joint sighthum workshop on computational linguistics for cultural heritage, social sciences, humanities and literature* (pp. 29–41).
- Anderson, P., He, X., Buehler, C., Teney, D., Johnson, M., Gould, S., & Zhang, L. (2018). Bottom-up and top-down attention for image captioning and visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 6077–6086).
- Antol, S., Agrawal, A., Lu, J., Mitchell, M., Batra, D., Zitnick, C. L., & Parikh, D. (2015). Vqa: Visual question answering. In *Proceedings of the IEEE international conference on computer vision* (pp. 2425–2433).
- Averill, J. R. (1999). Individual differences in emotional creativity: Structure and correlates. *Journal of personality*, 67(2), 331–371.
- Basu, K., Shakerin, F., & Gupta, G. (2020). Aqua: Asp-based visual question answering. In *International symposium on practical aspects of declarative languages* (pp. 57–72).
- Ben-Younes, H., Cadene, R., Cord, M., & Thome, N. (2017). Mutan: Multimodal tucker fusion for visual question answering. In *Proceedings of the IEEE international conference on computer vision* (pp. 2612–2620).
- Bertero, D., & Fung, P. (2016a). Deep learning of audio and language features for humor prediction. In *Proceedings of the tenth international conference on language resources and evaluation (lrec'16)* (pp. 496–501).
- Bertero, D., & Fung, P. (2016b). A long short-term memory framework for predicting humor in dialogues. In *Proceedings of the 2016 conference of the north american chapter of the association for computational linguistics: Human language technologies* (pp. 130–135).
- Binsted, K., & Ritchie, G. (1994). *An implemented model of punning riddles* (Tech. Rep.). University of Edinburgh. Department of Artificial Intelligence.
- Bongini, P., Becattini, F., Bagdanov, A. D., & Del Bimbo, A. (2020). Visual question answering for cultural heritage. In *Iop conference series: Materials science and engineering* (Vol. 949, p. 012074).
- Burns, A., Tan, R., Saenko, K., Sclaroff, S., & Plummer, B. A. (2019). Language features matter: Effec-

- tive language representations for vision-language tasks. In *Proceedings of the IEEE International Conference on Computer Vision* (pp. 7474–7483).
- Cao, Z., Hidalgo, G., Simon, T., Wei, S.-E., & Sheikh, Y. (2019). Openpose: realtime multi-person 2d pose estimation using part affinity fields. *IEEE transactions on pattern analysis and machine intelligence*, 43(1), 172–186.
- Chandrasekaran, A., Vijayakumar, A. K., Antol, S., Bansal, M., Batra, D., Zitnick, C. L., & Parikh, D. (2016). We are humor beings: Understanding and predicting visual humor. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 4603–4612).
- Chen, L., & Lee, C. M. (2017). Convolutional neural network for humor recognition. *arXiv preprint arXiv:1702.02584*.
- Chen, L., Li, Q., Wang, H., & Long, Y. (2018). Static correlative filter based convolutional neural network for visual question answering. In *2018 IEEE International Conference on Big Data and Smart Computing (BigComp)* (pp. 526–529).
- Chou, S.-H., Chao, W.-L., Lai, W.-S., Sun, M., & Yang, M.-H. (2020). Visual question answering on 360deg images. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision* (pp. 1607–1616).
- Clark, C., & Gardner, M. (2018). Simple and effective multi-paragraph reading comprehension. In *Proceedings of the 56th annual meeting of the association for computational linguistics (volume 1: Long papers)* (pp. 845–855).
- Colas, A., Kim, S., Dérmoncourt, F., Gupte, S., Wang, D. Z., & Kim, D. S. (2019). Tutorialvqa: Question answering dataset for tutorial videos. *arXiv preprint arXiv:1912.01046*.
- Conneau, A., Kiela, D., Schwenk, H., Barrault, L., & Bordes, A. (2017). Supervised learning of universal sentence representations from natural language inference data. In *Proceedings of the 2017 conference on empirical methods in natural language processing* (pp. 670–680).
- Constantin, M. G., Redi, M., Zen, G., & Ionescu, B. (2019). Computational understanding of visual interestingness beyond semantics: literature survey and analysis of covariates. *ACM Computing Surveys (CSUR)*, 52(2), 1–37.
- Czapla, B. F. P., & Howard, J. (2019). Applying a pre-trained language model to spanish twitter humor prediction. In *Proceedings of the Iberian Languages Evaluation Forum (IberLef 2019). CEUR Workshop Proceedings, CEUR-WS, Bilbao, Spain (9 2019)*.
- Dai, Z., Yang, Z., Yang, Y., Carbonell, J., Le, Q. V., & Salakhutdinov, R. (2019). Transformer-xl: Attentive language models beyond a fixed-length context. *arXiv preprint arXiv:1901.02860*.
- Das, A., Kottur, S., Gupta, K., Singh, A., Yadav, D., Moura, J. M., ... Batra, D. (2017). Visual dialog. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 326–335).
- Das, A., Kottur, S., Moura, J. M., Lee, S., & Batra, D. (2017). Learning cooperative visual dialog agents with deep reinforcement learning. In *Proceedings of the IEEE International Conference on Computer Vision* (pp. 2951–2960).
- Dasigi, P., Liu, N. F., Marasovic, A., Smith, N. A., & Gardner, M. (2019). Quoref: A reading comprehension dataset with questions requiring coreferential reasoning. In *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (emnlp-ijcnlp)* (pp. 5927–5934).
- Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., & Fei-Fei, L. (2009). Imagenet: A large-scale hierar-

- chical image database. In *2009 ieee conference on computer vision and pattern recognition* (pp. 248–255).
- Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.
- Di Gangi, M. A., Negri, M., & Turchi, M. (2019). Adapting transformer to end-to-end spoken language translation. In *Interspeech 2019* (pp. 1133–1137).
- Dybala, P., Ptaszynski, M., Higuchi, S., Rzepka, R., & Araki, K. (2008). Humor prevails!-implementing a joke generator into a conversational system. In *Australasian joint conference on artificial intelligence* (pp. 214–225).
- Engelthaler, T., & Hills, T. T. (2018). Humor norms for 4,997 english words. *Behavior research methods*, 50(3), 1116–1124.
- Fan, X., Lin, H., Yang, L., Diao, Y., Shen, C., Chu, Y., & Zou, Y. (2020). Humor detection via an internal and external neural network. *Neurocomputing*.
- Farazi, M., Khan, S., & Barnes, N. (2020). Attention guided semantic relationship parsing for visual question answering. *arXiv preprint arXiv:2010.01725*.
- Gan, Z., Gan, C., He, X., Pu, Y., Tran, K., Gao, J., ... Deng, L. (2017). Semantic compositional networks for visual captioning. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 5630–5639).
- Gao, H., Mao, J., Zhou, J., Huang, Z., Wang, L., & Xu, W. (2015). Are you talking to a machine? dataset and methods for multilingual image question answering. *arXiv preprint arXiv:1505.05612*.
- Gao, J., Ge, R., Chen, K., & Nevatia, R. (2018). Motion-appearance co-memory networks for video question answering. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 6576–6585).
- Garcia, N., Otani, M., Chu, C., & Nakashima, Y. (2020). KnowIT VQA: Answering knowledge-based questions about videos. In *Proceedings of the thirty-fourth aaai conference on artificial intelligence*.
- Garcia, N., Ye, C., Liu, Z., Hu, Q., Otani, M., Chu, C., ... Mitamura, T. (2020). A dataset and baselines for visual question answering on art. In *European conference on computer vision* (pp. 92–108).
- Gokhale, T., Banerjee, P., Baral, C., & Yang, Y. (2020). Vqa-lol: Visual question answering under the lens of logic. In *European conference on computer vision* (pp. 379–396).
- Gong, L., He, D., Li, Z., Qin, T., Wang, L., & Liu, T. (2019). Efficient training of BERT by progressively stacking. In *International conference on machine learning* (pp. 2337–2346).
- Goyal, Y., Khot, T., Summers-Stay, D., Batra, D., & Parikh, D. (2017). Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 6904–6913).
- Gurari, D., Li, Q., Stangl, A. J., Guo, A., Lin, C., Grauman, K., ... Bigham, J. P. (2018). Vizwiz grand challenge: Answering visual questions from blind people. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 3608–3617).
- Hasan, M. K., Rahman, W., Zadeh, A. B., Zhong, J., Tanveer, M. I., Morency, L.-P., & Hoque, M. E. (2019). Ur-funny: A multimodal language dataset for understanding humor. In *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (emnlp-ijcnlp)* (pp. 2046–2056).

- He, K., Zhang, X., Ren, S., & Sun, J. (2016a). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770–778).
- He, K., Zhang, X., Ren, S., & Sun, J. (2016b). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770–778).
- He, X., Zhang, Y., Mou, L., Xing, E., & Xie, P. (2020). Pathvqa: 30000+ questions for medical visual question answering. *arXiv preprint arXiv:2003.10286*.
- Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8), 1735–1780.
- Hossain, N., Krumm, J., & Gamon, M. (2019). "president vows to cut taxes; hair": Dataset and analysis of creative text editing for humorous headlines. *arXiv preprint arXiv:1906.00274*.
- Huang, D., Chen, P., Zeng, R., Du, Q., Tan, M., & Gan, C. (2020). Location-aware graph convolutional networks for video question answering. In *Proceedings of the AAAI conference on artificial intelligence* (Vol. 34, pp. 11021–11028).
- Huang, J., Li, Y., Tao, J., Lian, Z., Wen, Z., Yang, M., & Yi, J. (2017). Continuous multimodal emotion prediction based on long short term memory recurrent neural network. In *Proceedings of the 7th annual workshop on audio/visual emotion challenge* (pp. 11–18).
- Ionescu, C., Papava, D., Olaru, V., & Sminchisescu, C. (2013). Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. *IEEE transactions on pattern analysis and machine intelligence*, 36(7), 1325–1339.
- Jiang, Y.-G., Xu, B., & Xue, X. (2014). Predicting emotions in user-generated videos. In *Proceedings of the AAAI conference on artificial intelligence* (Vol. 28).
- Kayatani, Y., Yang, Z., Otani, M., Garcia, N., Chu, C., Nakashima, Y., & Takemura, H. (2021). The laughing machine: Predicting humor in video. In *The IEEE winter conference on applications of computer vision*.
- Kenfack, F. K., Siddiky, F. A., Balint-Benczedi, F., & Beetz, M. (2020). Robotvqa—a scene-graph-and deep-learning-based visual question answering system for robot manipulation. In *IEEE/RSJ international conference on intelligent robots and systems (IROS), Las Vegas, USA*.
- Kim, J., Ma, M., Kim, K., Kim, S., & Yoo, C. D. (2019). Progressive attention memory network for movie story question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 8337–8346).
- Kim, K.-M., Choi, S.-H., Kim, J.-H., & Zhang, B.-T. (2018). Multimodal dual attention memory for video story question answering. In *Proceedings of the European conference on computer vision (ECCV)* (pp. 673–688).
- Kim, K.-M., Heo, M.-O., Choi, S.-H., & Zhang, B.-T. (2017). Deepstory: Video story qa by deep embedded memory networks. *arXiv preprint arXiv:1707.00836*.
- Kiros, R., Zhu, Y., Salakhutdinov, R. R., Zemel, R., Urtasun, R., Torralba, A., & Fidler, S. (2015). Skip-thought vectors. In *Advances in neural information processing systems* (pp. 3294–3302).
- Kleinberg, B., van der Vegt, I., & Mozes, M. (2020). Measuring emotions in the covid-19 real world worry dataset. *arXiv preprint arXiv:2004.04225*.
- Krishna, R., Hata, K., Ren, F., Fei-Fei, L., & Carlos Nibbles, J. (2017). Dense-captioning events in videos. In *Proceedings of the IEEE international conference on computer vision* (pp. 706–715).
- Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P., & Soricut, R. (2019). Albert: A lite bert

- for self-supervised learning of language representations. In *International conference on learning representations*.
- Lei, J., Yu, L., Bansal, M., & Berg, T. (2018). Tvqa: Localized, compositional video question answering. In *Proceedings of the 2018 conference on empirical methods in natural language processing* (pp. 1369–1379).
- Lei, J., Yu, L., Berg, T. L., & Bansal, M. (2019). Tvqa+: Spatio-temporal grounding for video question answering. *arXiv preprint arXiv:1904.11574*.
- Lei, J., Yu, L., Berg, T. L., & Bansal, M. (2020a). Tvr: A large-scale dataset for video-subtitle moment retrieval. *arXiv preprint arXiv:2001.09099*.
- Lei, J., Yu, L., Berg, T. L., & Bansal, M. (2020b). Tvr: A large-scale dataset for video-subtitle moment retrieval. *arXiv preprint arXiv:2001.09099*.
- Li, L., Chen, Y.-C., Cheng, Y., Gan, Z., Yu, L., & Liu, J. (2020). Hero: Hierarchical encoder for video+ language omni-representation pre-training. *arXiv preprint arXiv:2005.00200*.
- Li, Y., Ouyang, W., Zhou, B., Wang, K., & Wang, X. (2017). Scene graph generation from objects, phrases and region captions. In *Proceedings of the IEEE international conference on computer vision* (pp. 1261–1270).
- Li, Z., Ding, X., & Liu, T. (2019). Story ending prediction by transferable bert. In *Proceedings of the 28th international joint conference on artificial intelligence* (pp. 1800–1806).
- Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., ... Zitnick, C. L. (2014). Microsoft coco: Common objects in context. In *European conference on computer vision* (pp. 740–755).
- Liu, B., Zhan, L.-M., Xu, L., Ma, L., Yang, Y., & Wu, X.-M. (2021). Slake: A semantically-labeled knowledge-enhanced dataset for medical visual question answering. In *2021 IEEE 18th international symposium on biomedical imaging (ISBI)* (pp. 1650–1654).
- Liu, C., Tang, T., Lv, K., & Wang, M. (2018). Multi-feature based emotion recognition for video clips. In *Proceedings of the 20th ACM international conference on multimodal interaction* (pp. 630–634).
- Liu, H., Gong, S., Ji, Y., Yang, J., Xing, T., & Liu, C. (2018). Multimodal cross-guided attention networks for visual question answering. In *2018 international conference on computer modeling, simulation and algorithm (CMSA 2018)*.
- Liu, M., Wang, X., Nie, L., He, X., Chen, B., & Chua, T.-S. (2018). Attentive moment retrieval in videos. In *The 41st international ACM SIGIR conference on research & development in information retrieval* (pp. 15–24).
- Liu, W., Angelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., & Berg, A. C. (2016). Ssd: Single shot multibox detector. In *European conference on computer vision* (pp. 21–37).
- Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., ... Stoyanov, V. (2019). Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.
- Lobry, S., Marcos, D., Murray, J., & Tuia, D. (2020). Rsvqa: Visual question answering for remote sensing data. *IEEE Transactions on Geoscience and Remote Sensing*, 58(12), 8555–8566.
- Lu, J., Batra, D., Parikh, D., & Lee, S. (2019). Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. In *Advances in neural information processing systems* (pp. 13–23).
- Lu, J., Xiong, C., Parikh, D., & Socher, R. (2017). Knowing when to look: Adaptive attention via a

- visual sentinel for image captioning. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 375–383).
- Lu, L., Liu, C., Li, J., & Gong, Y. (2020). Exploring transformers for large-scale speech recognition. *arXiv preprint arXiv:2005.09684*.
- Lu, P., Ji, L., Zhang, W., Duan, N., Zhou, M., & Wang, J. (2018). R-vqa: learning visual relation facts with semantic attention for visual question answering. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining* (pp. 1880–1889).
- Lucey, P., Cohn, J. F., Kanade, T., Saragih, J., Ambadar, Z., & Matthews, I. (2010). The extended cohn-kanade dataset (ck+): A complete dataset for action unit and emotion-specified expression. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition-Workshops* (pp. 94–101).
- Ma, X., Xu, P., Wang, Z., Nallapati, R., & Xiang, B. (2019). Universal text representation from bert: An empirical study. *arXiv preprint arXiv:1910.07973*.
- Mamgai, D., Brodiya, S., Yadav, R., & Dua, M. (2019). An improved automated question answering system from lecture videos. In *Proceedings of 2nd international conference on communication, computing and networking* (pp. 653–659).
- Marino, K., Rastegari, M., Farhadi, A., & Mottaghi, R. (2019). Ok-vqa: A visual question answering benchmark requiring external knowledge. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 3195–3204).
- Martinez, J., Hossain, R., Romero, J., & Little, J. J. (2017). A simple yet effective baseline for 3d human pose estimation. In *Iccv*.
- McCann, B., Bradbury, J., Xiong, C., & Socher, R. (2017). Learned in translation: Contextualized word vectors. In *Advances in neural information processing systems* (pp. 6294–6305).
- McCarthy, J. (2007). What is artificial intelligence.
- Meyer, J. C. (2000). Humor as a double-edged sword: Four functions of humor in communication. *Communication theory*, 10(3), 310–331.
- Mittal, T., Guhan, P., Bhattacharya, U., Chandra, R., Bera, A., & Manocha, D. (2020). Emoticon: Context-aware multimodal emotion recognition using Frege’s principle. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 14234–14243).
- Mun, J., Cho, M., & Han, B. (2020). Local-global video-text interactions for temporal grounding. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 10810–10819).
- Mun, J., Hongsuck Seo, P., Jung, I., & Han, B. (2017). Marioqa: Answering questions by watching gameplay videos. In *Proc. iccv* (pp. 2867–2875).
- Murarka, A., Radhakrishnan, B., & Ravichandran, S. (2020). Detection and classification of mental illnesses on social media using roberta. *arXiv preprint arXiv:2011.11226*.
- Narasimhan, M., Lazebnik, S., & Schwing, A. G. (2018). Out of the box: Reasoning with graph convolution nets for factual visual question answering. *arXiv preprint arXiv:1811.00538*.
- Narasimhan, M., & Schwing, A. G. (2018). Straight to the facts: Learning knowledge base retrieval for factual visual question answering. In *Proceedings of the European conference on computer vision (ECCV)* (pp. 451–468).
- Ogawa, T., Sasaka, Y., Maeda, K., & Haseyama, M. (2018). Favorite video classification based on

- multimodal bidirectional lstm. *IEEE Access*, 6, 61401–61409.
- Patil, N., & Biswas, P. K. (2016). Global abnormal events detection in surveillance video—a hierarchical approach. In *2016 sixth international symposium on embedded computing and system design (ised)* (pp. 217–222).
- Patro, B. N., Lunayach, M., Srivastava, D., Singh, H., Namboodiri, V. P., et al. (2021). Multimodal humor dataset: Predicting laughter tracks for sitcoms. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision* (pp. 576–585).
- Pennington, J., Socher, R., & Manning, C. (2014). Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing* (pp. 1532–1543).
- Pestian, J. P., Matykiewicz, P., Linn-Gust, M., South, B., Uzuner, O., Wiebe, J., ... Brew, C. (2012). Sentiment analysis of suicide notes: A shared task. *Biomedical informatics insights*, 5, BII–S9042.
- Rajpurkar, P., Jia, R., & Liang, P. (2018). Know what you don't know: Unanswerable questions for squad. In *Proceedings of the 56th annual meeting of the association for computational linguistics (volume 2: Short papers)* (pp. 784–789).
- Ren, M., Kiros, R., & Zemel, R. (2015). Image question answering: A visual semantic embedding model and a new dataset. *Proc. Advances in Neural Inf. Process. Syst.*, 1(2), 5.
- Ruwa, N., Mao, Q., Wang, L., Gou, J., & Dong, M. (2019). Mood-aware visual question answering. *Neurocomputing*, 330, 305–316.
- Saffar Mehrjardi, M. (2019). Self-attentional models application in task-oriented dialogue generation systems.
- Shah, S., Mishra, A., Yadati, N., & Talukdar, P. P. (2019). Kvqa: Knowledge-aware visual question answering. In *Proceedings of the aaai conference on artificial intelligence* (Vol. 33, pp. 8876–8884).
- Shen, G., Wang, X., Duan, X., Li, H., & Zhu, W. (2020). Memor: A dataset for multimodal emotion reasoning in videos. In *Proceedings of the 28th acm international conference on multimedia* (pp. 493–502).
- Sjöbergh, J., & Araki, K. (2009). A measure of funniness, applied to finding funny things in wordnet. In *Proceedings of the conference of the pacific association for computational linguistics 2009* (pp. 236–241).
- Socher, R., Perelygin, A., Wu, J., Chuang, J., Manning, C. D., Ng, A., & Potts, C. (2013). Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 conference on empirical methods in natural language processing* (pp. 1631–1642).
- Song, X., Shi, Y., Chen, X., & Han, Y. (2018). Explore multi-step reasoning in video question answering. In *Proceedings of the 26th acm international conference on multimedia* (pp. 239–247).
- Stock, O., & Strapparava, C. (2005). Hahacronym: A computational humor system. In *Proceedings of the acl interactive poster and demonstration sessions* (pp. 113–116).
- Su, W., Zhu, X., Cao, Y., Li, B., Lu, L., Wei, F., & Dai, J. (2019). Vi-bert: Pre-training of generic visual-linguistic representations. *arXiv preprint arXiv:1908.08530*.
- Sun, C., Myers, A., Vondrick, C., Murphy, K., & Schmid, C. (2019). Videobert: A joint model for video and language representation learning. In *Proceedings of the IEEE/CVF international conference on computer vision* (pp. 7464–7473).

- Tan, L., Zhang, K., Wang, K., Zeng, X., Peng, X., & Qiao, Y. (2017). Group emotion recognition with individual facial emotion cnns and global image based cnns. In *Proceedings of the 19th acm international conference on multimodal interaction* (pp. 549–552).
- Tapaswi, M., Zhu, Y., Stiefelhagen, R., Torralba, A., Urtasun, R., & Fidler, S. (2016a). Movieqa: Understanding stories in movies through question-answering. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 4631–4640).
- Tapaswi, M., Zhu, Y., Stiefelhagen, R., Torralba, A., Urtasun, R., & Fidler, S. (2016b). Movieqa: Understanding stories in movies through question-answering. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 4631–4640).
- Turian, J., Ratinov, L., & Bengio, Y. (2010). Word representations: a simple and general method for semi-supervised learning. In *Proceedings of the 48th annual meeting of the association for computational linguistics* (pp. 384–394).
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., . . . Polosukhin, I. (2017). Attention is all you need. In *Proc. nips* (pp. 5998–6008).
- Vig, J. (2019, July). A multiscale visualization of attention in the transformer model. In *Proceedings of the 57th annual meeting of the association for computational linguistics: System demonstrations* (pp. 37–42). Florence, Italy: Association for Computational Linguistics. Retrieved from <https://www.aclweb.org/anthology/P19-3007> doi: 10.18653/v1/P19-3007
- Wang, J., Jiang, W., Ma, L., Liu, W., & Xu, Y. (2018). Bidirectional attentive fusion with context gating for dense video captioning. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 7190–7198).
- Wang, P., Wu, Q., Shen, C., Dick, A., & Van Den Hengel, A. (2017). Fvqa: Fact-based visual question answering. *IEEE transactions on pattern analysis and machine intelligence*, 40(10), 2413–2427.
- Wang, P., Wu, Q., Shen, C., Hengel, A. v. d., & Dick, A. (2015). Explicit knowledge-based reasoning for visual question answering. *arXiv preprint arXiv:1511.02570*.
- Weller, O., Fulda, N., & Seppi, K. (2020). Can humor prediction datasets be used for humor generation? humorous headline generation via style transfer. In *Proceedings of the second workshop on figurative language processing* (pp. 186–191).
- Weller, O., & Seppi, K. (2019). Humor detection: A transformer gets the last laugh. In *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (emnlp-ijcnlp)* (pp. 3612–3616).
- Winterbottom, T., Xiao, S., McLean, A., & Al Moubayed, N. (2020). On modality bias in the tvqa dataset.
- Winters, T., Nys, V., & De Schreye, D. (2018). Automatic joke generation: Learning humor from examples. In *International conference on distributed, ambient, and pervasive interactions* (pp. 360–377).
- Winters, T., Nys, V., & De Schreye, D. (2019). Towards a general framework for humor generation from rated examples. http://computationalcreativity.net/iccc2019/assets/iccc_proceedings_2019.pdf (Proceedings of the 10th International Conference on Computational Creativity), 274–281.
- Wu, Q., Wang, P., Shen, C., Dick, A., & Van Den Hengel, A. (2016). Ask me anything: Free-form visual question answering based on knowledge from external sources. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 4622–4630).

- Wu, Y.-C., Chang, C.-H., & Lee, Y.-S. (2004). Clvq: Cross-language video question/answering system. In *Ieee sixth international symposium on multimedia software engineering* (pp. 294–301).
- Wu, Y.-C., & Yang, J.-C. (2008). A robust passage retrieval algorithm for video question answering. *IEEE Transactions on Circuits and Systems for Video Technology*, 18(10), 1411–1421.
- Xi, Y., Zhang, Y., Ding, S., & Wan, S. (2020). Visual question answering model based on visual relationship detection. *Signal Processing: Image Communication*, 80, 115648.
- Xu, H., & Saenko, K. (2016). Ask, attend and answer: Exploring question-guided spatial attention for visual question answering. In *European conference on computer vision* (pp. 451–466).
- Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A., Salakhudinov, R., ... Bengio, Y. (2015). Show, attend and tell: Neural image caption generation with visual attention. In *International conference on machine learning* (pp. 2048–2057).
- Xu, Z. (2021). Roberta-wwm-ext fine-tuning for chinese text classification. *arXiv preprint arXiv:2103.00492*.
- Yang, A., Miech, A., Sivic, J., Laptev, I., & Schmid, C. (2020). Just ask: Learning to answer questions from millions of narrated videos. *arXiv preprint arXiv:2012.00451*.
- Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhudinov, R. R., & Le, Q. V. (2019). Xlnet: Generalized autoregressive pretraining for language understanding. In *Advances in neural information processing systems* (pp. 5754–5764).
- Yao, T., Pan, Y., Li, Y., & Mei, T. (2018). Exploring visual relationship for image captioning. In *Proceedings of the european conference on computer vision (eccv)* (pp. 684–699).
- Ye, Y., Zhao, Z., Li, Y., Chen, L., Xiao, J., & Zhuang, Y. (2017). Video question answering via attribute-augmented attention network learning. In *Proceedings of the 40th international acm sigir conference on research and development in information retrieval* (pp. 829–832).
- Yin, X., & Ordonez, V. (2017). Obj2text: Generating visually descriptive language from object layouts. In *Proceedings of the 2017 conference on empirical methods in natural language processing* (pp. 177–187).
- You, Q., Luo, J., Jin, H., & Yang, J. (2016). Building a large scale dataset for image emotion recognition: The fine print and the benchmark. In *Proceedings of the aaai conference on artificial intelligence* (Vol. 30).
- Yu, D., Gao, X., & Xiong, H. (2018). Structured semantic representation for visual question answering. In *2018 25th ieee international conference on image processing (ICIP)* (pp. 2286–2290).
- Yu, Y., Kim, J., & Kim, G. (2018). A joint sequence fusion model for video question answering and retrieval. In *Proceedings of the european conference on computer vision (eccv)* (pp. 471–487).
- Yu, Y., Ko, H., Choi, J., & Kim, G. (2017). End-to-end concept word detection for video captioning, retrieval, and question answering. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 3165–3173).
- Zareian, A., Karaman, S., & Chang, S.-F. (2020). Weakly supervised visual semantic parsing. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 3736–3745).
- Zeng, K.-H., Chen, T.-H., Chuang, C.-Y., Liao, Y.-H., Niebles, J. C., & Sun, M. (2017). Leveraging video descriptions to learn video question answering. In *Proceedings of the aaai conference on artificial intelligence* (Vol. 31).

- Zhang, H., Liu, D., Lv, J., & Luo, C. (2020). Let's be humorous: Knowledge enhanced humor generation. *arXiv preprint arXiv:2004.13317*.
- Zhang, Z., Wu, Y., Zhao, H., Li, Z., Zhang, S., Zhou, X., & Zhou, X. (2020). Semantics-aware bert for language understanding. In *Proceedings of the aaai conference on artificial intelligence* (Vol. 34, pp. 9628–9635).
- Zhao, Z., Jiang, X., Cai, D., Xiao, J., He, X., & Pu, S. (2018). Multi-turn video question answering via multi-stream hierarchical attention context network. In *Ijcai* (pp. 3690–3696).
- Zhao, Z., Yang, Q., Cai, D., He, X., & Zhuang, Y. (2017). Video question answering via hierarchical spatio-temporal attention networks. In *Ijcai* (pp. 3518–3524).
- Zhong, W., Xu, J., Tang, D., Xu, Z., Duan, N., Zhou, M., ... Yin, J. (2019). Reasoning over semantic-level graph for fact checking. *arXiv preprint arXiv:1909.03745*.
- Zhu, J., Xia, Y., Wu, L., He, D., Qin, T., Zhou, W., ... Liu, T.-Y. (2020). Incorporating bert into neural machine translation. *arXiv preprint arXiv:2002.06823*.
- Zhu, L., Xu, Z., Yang, Y., & Hauptmann, A. G. (2017). Uncovering the temporal context for video question answering. *International Journal of Computer Vision*, 124(3), 409–421.
- Zhu, Y., Groth, O., Bernstein, M., & Fei-Fei, L. (2016). Visual7w: Grounded question answering in images. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 4995–5004).
- Zhu, Y., Kiros, R., Zemel, R., Salakhutdinov, R., Urtasun, R., Torralba, A., & Fidler, S. (2015). Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *Proceedings of the ieee international conference on computer vision* (pp. 19–27).
- Zhu, Y., Lim, J. J., & Fei-Fei, L. (2017). Knowledge acquisition for visual question answering via iterative querying. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 1154–1163).