이학석사 학위논문

Mounting Visual Metadata on Transformer-based Language Model for Open-ended Video Question Answering

비디오 메타데이터를 활용한 트랜스포머 기반 주관식 비디오 질의 응답

2023년 8월

서울대학교 대학원

협동과정 인지과학전공

이 동 건

Mounting Visual Metadata on Transformer-based Language Model for Open-ended Video Question Answering

비디오 메타데이터를 활용한 트랜스포머 기반 주관식 비디오 질의 응답

지도 교수 장 병 탁

이 논문을 이학석사 학위논문으로 제출함
2023년 7월

서울대학교 대학원
협동과정 인지과학전공
이 동 건

이동건의 이학석사 학위논문을 인준함
2023년 7월

위 원 장 ______ 고 성 룡 ______ (인)

부위원장 ______ 장 병 탁 ______ (인)

위 원 ______ 권 가 진 ______ (인)

Abstract

Mounting Visual Metadata on Transformer-based Language Model for Open-ended Video Question Answering

Donggeon Lee
Interdisciplinary Program in Cognitive Science
The Graduate School
Seoul National University

Video question answering has recently received a lot of attention from multimodal video researchers. Most video question answering datasets are usually in the form of multiple-choice. But, the model for the multiple-choice task does not infer the answer. Rather it compares the answer candidates for picking the correct answer. This method is limited in options, making it difficult to grasp detailed interactions between videos and questions. On the other hand, in the case of open-ended answer, it is easy for the model to understand the complex relationship between the video and the question through free answer generation. In addition, from a practical point of view, for interaction with humans, subjective interaction is easier than the method of providing answer candidates. In this paper, we challenge the existing multiple-choice video question answering by changing it to open-ended video question answering. To tackle open-ended question answering, we use the pretrained GPT2 model. In order to understand the contents of the video, information about the characters and events is needed. To utilize the aforementioned information, fine-tuning is performed using information such as video input, subtitles, metadata, and description. This study is performed by changing the existing DramaQA dataset to an open-ended question answering, and it shows that performance can be improved using video metadata.

Keywords : Video, VQA, open-ended, transformer, visual metadata, Language Model
Student Number : 2021-27378

Contents

Table Contents

Figure Contents

Chapter 1

Introduction

Transformers are now the de facto standard for language modeling and recently extending their applications in vision and multimodal domain [19, 4]. Transformers in the vision and language domain are usually pretrained with large scale datasets and applied to various downstream tasks. Among downstream tasks, video question answering evaluates whether the model understands various dimensions of video contents and is usually done in multiple-choice. However, when learning a model for multiple-choice video question answering, the model selects the correct answer by comparing the similarity between the question and the answer candidates rather than inferring the correct answer to the question. But, selecting the correct answer through comparison with the answer candidates does not perform the reasoning required in the question and answering, making it difficult to generalize for other tasks.

In this paper, we tackle the current multiple-choice video question answering dataset by changing it into an open-ended format. We focus on a more challenging open-ended setting where there is no prior knowledge of answer choices. As well as, in the case of open-ended VQA, additional data like multiple choices is not required to generate answers to new questions.

The answer candidates are not given in open-ended multimodal video question answering, so the model infers the correct answer through reasoning. In other words, in the case of multiple choice, among the candidates for the correct answer, one that is close to what the model understands is found. On the other hand, in the open-ended model, the model directly finds the answer to the question. In the case of open-ended model, through free answer generation, the model can more deeply understand the complex relationship between images and questions. In the case of multiple-choice VQA, it can be difficult to capture the detailed interactions between images and questions due to the limited number of choices.

In addition, open-ended expression is easy when interacting with humans in a practical aspect. In the real world, you can't always give 5 options. For example, even when used as an assistive technology for the visually impaired, it is difficult to give multiple choice options for sights that the disabled cannot see.

Challenging open-ended multimodal video question answering, we propose an extended model that learns various modalities together based on the recently proposed Transformer language model. The proposed model receives various metadata and language input of video. The results show that performance can be improved by combining multiple metadata rather than features from raw videos.

This paper is organized as follows. Chapter 2 examines related works to video question answering and open-ended question answering. Chapter 3 describes the proposed model and learning strategy. Chapter 4 examines the dataset and experimental settings, as well as the quantitative results. Finally, in Chapter 5, the conclusion and future research directions are described.

Chapter 2

Related Works

2.1 Video Question Answering

A variety of video question-answering datasets have been proposed, including MovieQA[17], PororoQA[10], TGIF-QA[9], TVQA[11], DramaQA[5], and are mostly in the multiple-choice format. AVSD Dataset[1] is characterized by the fact that question-answering for video is in the form of dialogue, which is out of the existing multiple-choice form.

Recently, various approaches have been proposed for video story question answering, which can be divided into three categories. There are techniques using Memory Network[17, 10], Attention[10, 11], and Transformer[21]. Memory networks stores and utilizes key information about a question-answering in a memory network to find it among many information in a long video. Attention effectively represents only the representation of visual/verbal core information by progressing attention across layers. Techniques utilizing context matching by applying attention achieved high performance in question-and-answer by comparing the context of a question-and-answer with the context of a given video in detail. Recently, researchers propose transformer-based models for video question answering. [18] proposed transformer and the proposed architecture brought a huge performance improvement in language modeling, and there is a move to expand it to a video domain. Recent state-of-art models show that these techniques can perform well in modeling the video as well as the language.

2.2 Open-Ended Question Answering

In the H. Xue et al.[20], Z. Zhao et al.[23], pointed out that the existing video question answering task used only one static image and text and also dealt with it as a short word oriented multiple-choice problem. It is emphasized that this approach cannot utilize the sequential and temporal information of the video. Therefore, its usability is limited in that the answer is chosen within given answers. In the above papers, the sequential/time information of the video was utilized to finally generate answers through decoders, resulting in better results than traditional methods (Mean-VQA, SS-VQA, etc.). However, the issues addressed by the above papers are limited in that they are short lived, although open-ended, and the format of questions and answers is also simple.

In the [12], the author conducted a study on AVSD task[1](Given video and ten turns of question answering a text, task generates natural language answers to the last question) based on Transformer(GPT2[15]). This paper extracts features from video and text with I3D[3] and VGGish[7], applies positional encoding, Beam Search, receives good results from several metrics (BLEU, METEOR, CIDEr, etc.). However, the model is not much different from above papers, and the position and video feature information was not used properly.

Chapter 3

Method

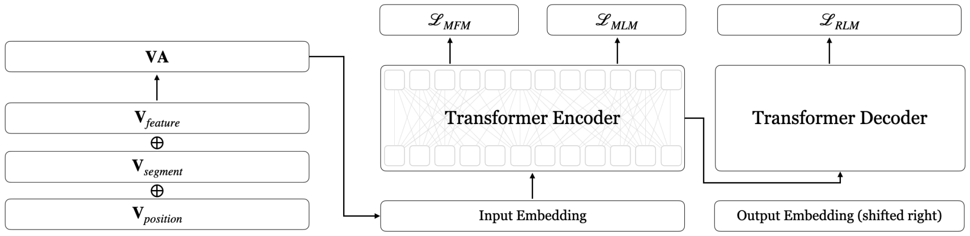


Figure 3.1 Overview of the proposed multimodal transformer model architecture. V_feature ： feature vector, V_segment ： segment tokens, V_position ： position encoding, MFM: Masked Frame Modeling, MLM ： Masked Language Modeling, RLM ： Response Language Modeling

3.1 Formulation

The purpose of our model is to integrate multimodal information (e.g., subtitle, video, audio, question, etc.) to generate the open-ended answer.

Our model consists of inputs of video, question and outputs of answer. The video is represented as V = ({v1,...,vN},{m1,...,mN},{s1,...,sM}). v_n is representing the n-th frame in V, m_n means a image features, and a visual meta data, the information such as person, person’s emotion and behavior, in bounding box corresponding to n-th frame, s_m is m-th subtitle in the entire video V. The question is represented as Q = $\{w_a^1, . . . , w_q^L\}$, and the answer is represented as A = $\{w_a^1,...,w_a^K\}$.

Each frame can be expressed as vv_n by extracting 3 frames per second from video and then feeding in the pre-trained I3D[3] model to extract feature vectors. There is information about the character in the form of $\{c^1 ,...,c^{Imn} \}$ in each mn. and information about each character is represented as $c^i = (f^i , p^i , b^i , e^i)$.

f_m^i is a feature representation of the character's image of bounding box using a pre-trained ResNet152[6] model. p^i_{mn} is a word embedding representation using a pre-trainned GPT2 model. b^i_{mn} is the character's behavior. e^i_{mn} is a word embedding representation of the character's emotion.

Each s an be expressed as $(p ,\{w^1 ,...,w^{Jsm} \})$ which which can be divided into sentence, $\{w^1 ,...,w^{Jsm}\}$, which can be divided into a word w^j and a speaker p Both speakers and words can be expressed in a previous way. Sentences can also be broken down into words using the GPT2 tokenizer.

3.2 GPT2

We reference and use GPT2, a transformer model, which uses attention in place of the previous recurrence and convolution based architectures. Attention mechanisms allow the model to selectively focus on segments of input text it predicts to be the most relevant.

GPT2 models receive the feature, segment, and position as inputs. Feature refers to data that embeds text input through GPT2 tokenizer, segment refers to data that means a token type of each word, such as [eos] and [sos], and position refers to the location of each word in the sentence.

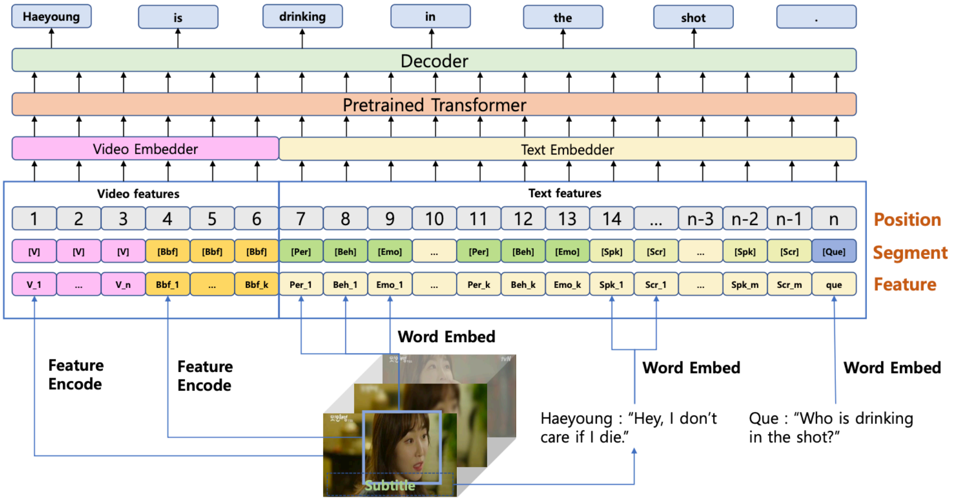


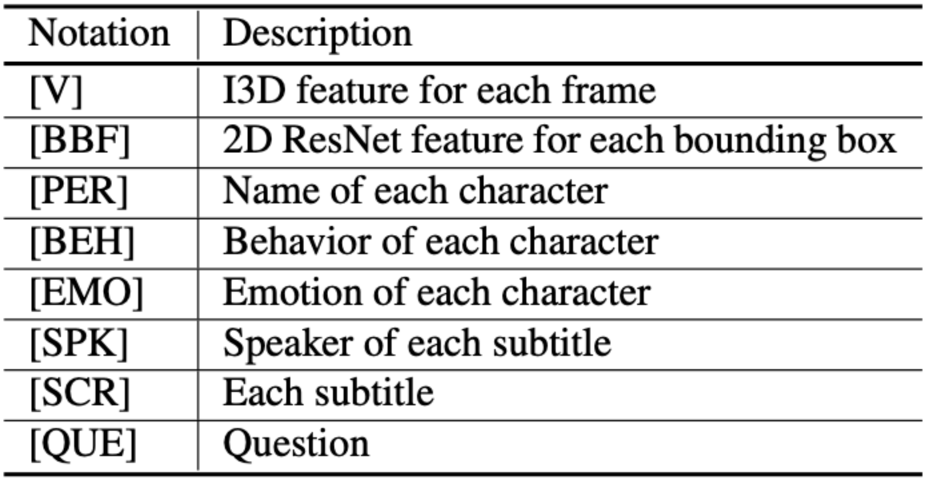
Figure 3.2 Multimodal transformer model architecture. The video embedder is a linear layer which embeds feature of video size to feature of embedding size, and the text embedder is a linear which embeds feature of vocab size to feature of embedding size. Denot We used the following segment tokens [V] : Video, [Bbf] : feature of bounding box, [Per] : person's name, [Beh] : person's behavior, [Emo] : person's emotion, [Spk] : speaker, [Scr] : script, [Que] : question.

3.2.1 Feature Embedding

Feature embedding input is all of the preceding (v , {c^{1} , . . . , c^{Imn} }) to a two-dimensional sequence over time. Subsequent (p , {w^{1} , . . . , w^{Jsm} }) similarly leads to a two-dimensional sequence over time. Finally, we attach {$w_q{}^{1}$, . . . , $w_q{}^{L}$ }. Therefore, the sequence length is N + $\Sigma^{N}{}_{mn=1}Imn$ + M + $\Sigma^{M}{}_{sm=1}Jsm$ + L. On the other hand, if features are extracted using I3D or ResNet, the features are different from those extracted with GPT2 models, so the dimensions are adjusted through a layer of learnable linear layers.

$$
\begin{aligned}
\mathbf{V}_{feature} = [&\{(v_{\mathbf{v}_n}, \{\mathbf{c}^{1}_{\mathbf{m}_n}, \ldots, \mathbf{c}^{I_{\mathbf{m}_n}}_{\mathbf{m}_n}\})\},\\
&\{(p_{\mathbf{s}_m}, \{w^{1}_{\mathbf{s}_m}, \ldots, w^{J_{\mathbf{s}_m}}_{\mathbf{s}_m}\})\},\\
&\{qw_1, \ldots, qw_L\}]
\end{aligned}
$$

3.2.2 Segment Embedding



Notation	Description
[V]	I3D feature for each frame
[BBF]	2D ResNet feature for each bounding box
[PER]	Name of each character
[BEH]	Behavior of each character
[EMO]	Emotion of each character
[SPK]	Speaker of each subtitle
[SCR]	Each subtitle
[QUE]	Question

Table 3.1 Notation and description for segments

Segment embedding distinguishes the various inputs that enter the video. The distinguishing features can be divided into eight as Table 3.1.

For each of these eight Feature categories, Segment embedding was performed using special token in GPT2.

3.3 Decoding Method

3.3.1 Beam Search

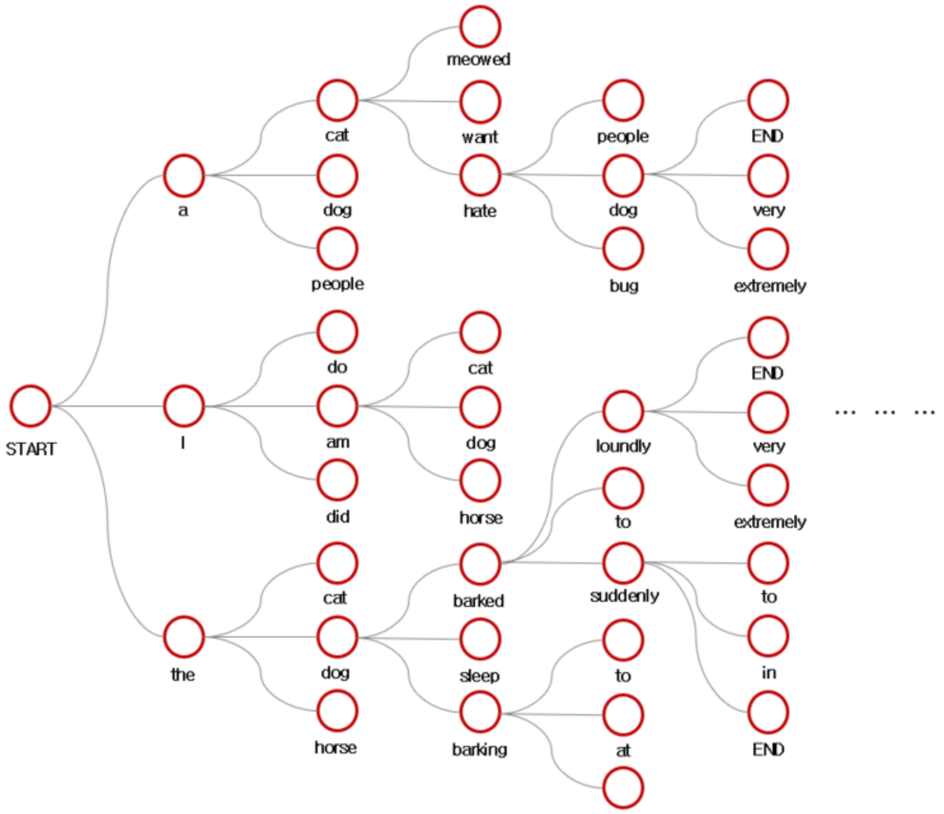


Figure 3.3 Beam search

Beam search is a heuristic search algorithm that explores a graph by expanding the most promising node in a limited set. Beam search is an optimization of best-first search that reduces its memory requirements. Best-first search is a graph search which orders all partial solutions according to some heuristic. But in beam search, only a predetermined number of best partial solutions are kept as candidates. It is thus a greedy algorithm.

3.3.2 Nucleus Sampling

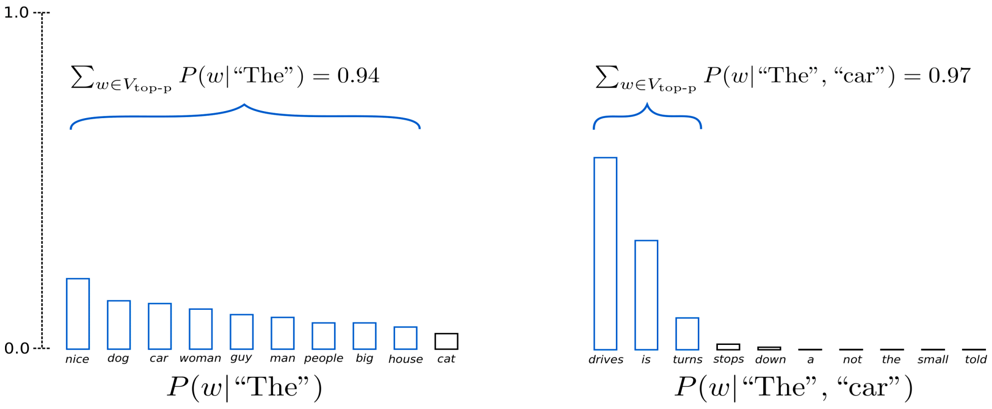


Figure 3.4 Nucleus Sampling

Deterministic sentence generation methods such as Beam Search, Greedy Search have the disadvantage of generating repeated words consecutively or resulting in too general sentences when used in an open-ended text generation task, which is a sentence generation task in which the end of a sentence is not determined.

Human generated sentences don't look like this. Therefore, in open-ended generation, a stochastic sampling-based sentence generation method, rather than a deterministic method, is often used to mimic the human sentence generation method.

Nucleus sampling is a method that compensates for the disadvantages of “Sampling with SoftMax temperature” and “Top-k sampling”, which are representative sampling methods. Nucleus sampling sorts the words in descending order of probability when the model calculates the probability of the next word to appear, and selects words in order until the point when the probability value of each word is accumulated exceeds the hyperparameter p. We then renormalize the probabilities of the selected words and sample words from that distribution.

Method	Bleu	Meteor	Bertscore	Bleurt	Time
Beam	0.69	0.2	0.34	0.62	130 min
Nucleus	0.68	0.18	0.32	0.6	8 min

Table 3.2 It is a description of the performance and time required for each Decoding Method for 3453 data in a subtitle-only environment.

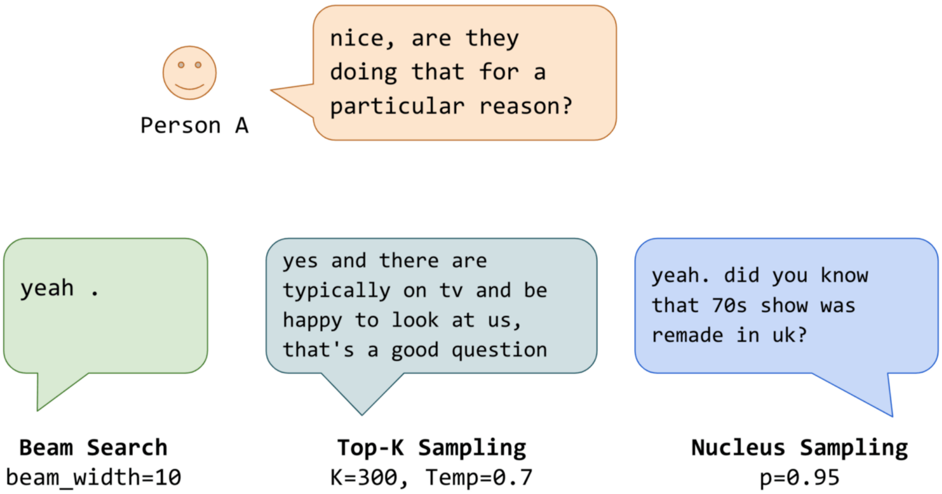

Person A
nice, are they doing that for a particular reason?
yeah .
yes and there are typically on tv and be happy to look at us, that's a good question
yeah. did you know that 70s show was remade in uk?
Beam Search
beam_width=10
Top-K Sampling
K=300, Temp=0.7
Nucleus Sampling
p=0.95

Figure 3.5 Decoding Strategy Comparison

To find an effective decoding method for multimodal answer generation, we try the decoding methods, including beam search and Nucleus Sampling[8] which samples text from the dynamic nucleus of the probability distribution. Although beam search showed slightly high performance, it took about 16 times more time to use it in real-time, so Neclues Sampling was used.

3.4 Implementation Details

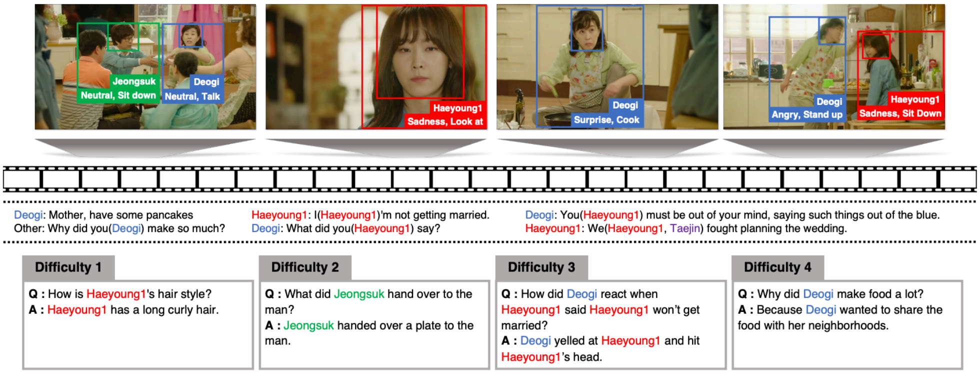
All experiments are run on NVIDIA [TITAN Xp]. Because of the lack of memory, we use a batch size of 1 input unit. We use AdamW optimizer[13] with a learning rate of 1e-4 and weight decay of 1e-5. Cross-entropy loss is used to train the model.

Chapter 4

Result

4.1 Settings

4.1.1 Dataset



Deogi: Mother, have some pancakes
Other: Why did you(Deogi) make so much?

Haeyoung1: I(Haeyoung1)'m not getting married.
Deogi: What did you(Haeyoung1) say?

Deogi: You(Haeyoung1) must be out of your mind, saying such things out of the blue.
Haeyoung1: We(Haeyoung1, Taejin) fought planning the wedding.

Q : How is Haeyoung1's hair style?
A : Haeyoung1 has a long curly hair.

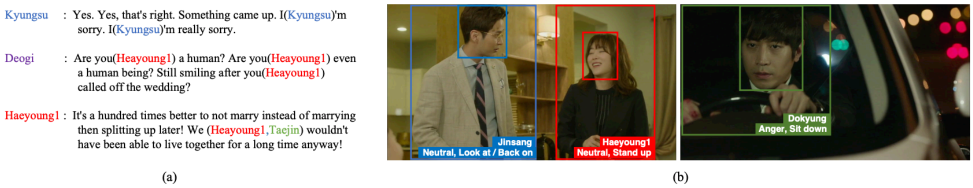
Q : What did Jeongsuk hand over to the man?
A : Jeongsuk handed over a plate to the man.

Q : How did Deogi react when Haeyoung1 said Haeyoung1 won't get married?
A : Deogi yelled at Haeyoung1 and hit Haeyoung1's head.

Q : Why did Deogi make food a lot?
A : Because Deogi wanted to share the food with her neighborhoods.

Figure 4.1 An example of DramaQA dataset which contains video clips, scripts, and QA pairs with levels of difficulty. A pair of QA corresponds to either a shot or a scene, and each QA is assigned one out of possible four stages of difficulty. A video clip consists of a sequence of images with visual annotations centering the main characters.

To show the effectiveness of the proposed method, we evaluate it on video question answering datasets, i.e., DramaQA [5]. This dataset is for multiple-choice tasks. So, the sentence corresponding to the correct answer among the multiple-choice options was converted into a label and used.

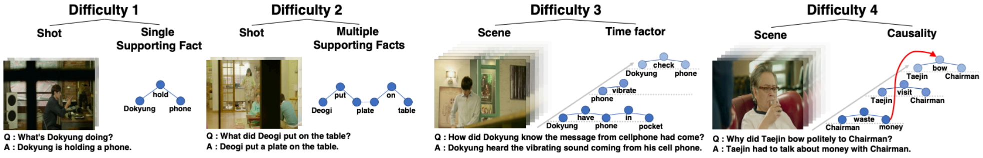


Kyungsu : Yes. Yes, that's right. Something came up. I(Kyungsu)'m sorry. I(Kyungsu)'m really sorry.

Deogi : Are you(Heayoung1) a human? Are you(Heayoung1) even a human being? Still smiling after you(Heayoung1) called off the wedding?

Haeyoung1 : It's a hundred times better to not marry instead of marrying then splitting up later! We (Heayoung1,Taejin) wouldn't have been able to live together for a long time anyway!

Figure 4.2 Examples of character-centered video annotations: (a) coreference resolved scripts and (b) visual metadata which contains the main characters' bounding box, name, behavior, and emotion. All annotations for characters in script and visual metadata can be co-referred by unique character's name.



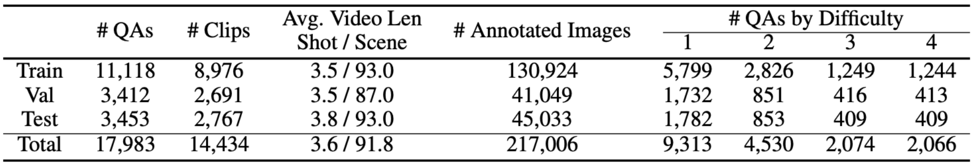
Q : What's Dokyung doing?
A : Dokyung is holding a phone.

Q : What did Deogi put on the table?
A : Deogi put a plate on the table.

Q : How did Dokyung know the message from cellphone had come?
A : Dokyung heard the vibrating sound coming from his cell phone.

Q : Why did Taejin bow politely to Chairman?
A : Taejin had to talk about money with Chairman.

Figure 4.3 Four examples of different QA level. Difficulty 1 and 2 target shot-length videos. Difficulty 1 requires single supporting fact to answer, and Difficulty 2 requires multiple supporting facts to answer. Difficulty 3 and 4 require a time factor to answer and target scene-length videos. Especially, Difficulty 4 requires causality between supporting facts from different time.



	# QAs	# Clips	Avg. Video Len Shot / Scene	# Annotated Images	# QAs by Difficulty			
					1	2	3	4
Train	11,118	8,976	3.5 / 93.0	130,924	5,799	2,826	1,249	1,244
Val	3,412	2,691	3.5 / 87.0	41,049	1,732	851	416	413
Test	3,453	2,767	3.8 / 93.0	45,033	1,782	853	409	409
Total	17,983	14,434	3.6 / 91.8	217,006	9,313	4,530	2,074	2,066

Table 4.1 Statistics about train, validation, and test split of DramaQA dataset. # QAs: The number of QA pairs. # Clips: The number of video clips including shot and scene. Avg. Video Len: Average video length per each video clip. # Annotated Images: The number of annotated images in total target video. # QAs by Difficulty: The number of QA pairs for each difficulty level.

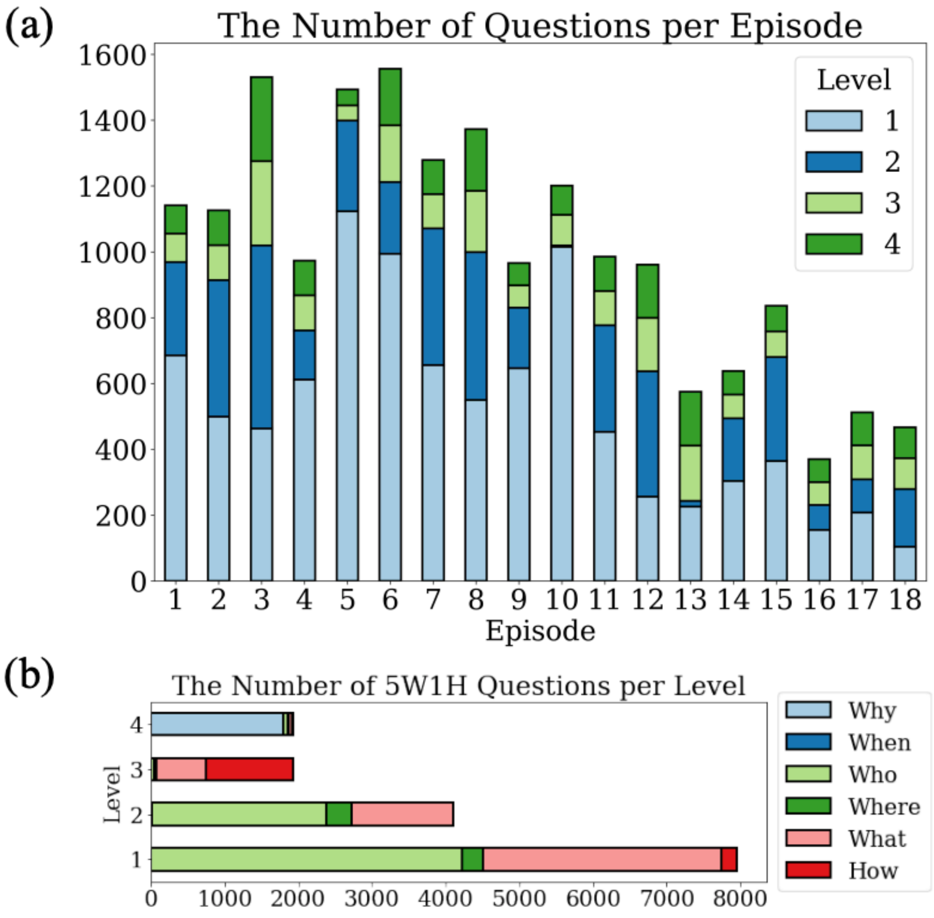


Figure 4.4 (a) The number of QA pairs per episode and difficulty level. Given that the length of scene is tens of times longer than the size of shot, the variation between levels is small compared to the number of videos. (b) The number of 5W1H question types per difficulty level.

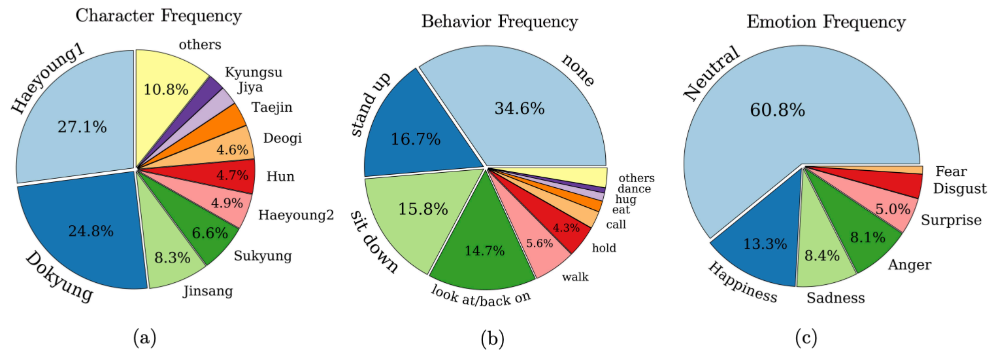
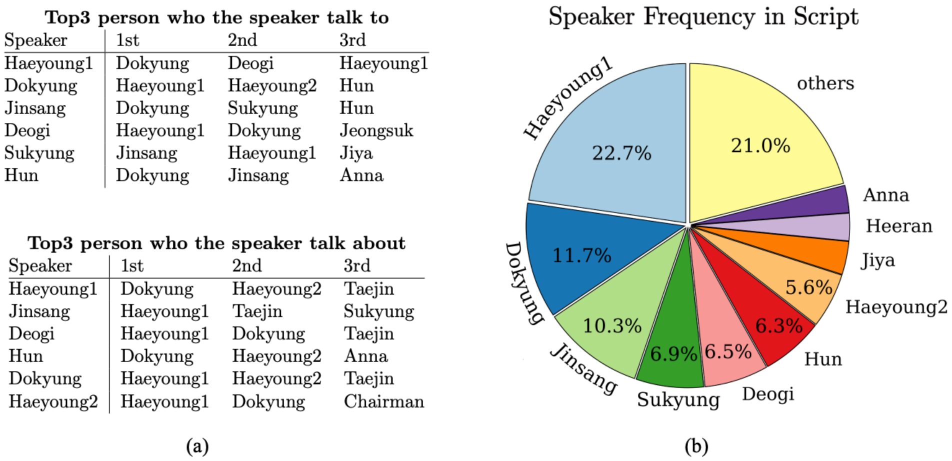


Figure 4.5 (a) The percentage of each character's frequency in visual metadata. Haeyoung1 and Dokyung are two main characters of drama AnotherMissOh. Haeyoung2 is the person who has same name with Haeyoung1, but we divided their name with numbers to get rid of confusion. (b) The percentage of each behavior frequency in the visual metadata. none behavior occupies a lot because there are many frames with only character's face. (c) The percentage of each emotion frequency in the visual metadata.



Top3 person who the speaker talk to

Speaker	1st	2nd	3rd
Haeyoung1	Dokyung	Deogi	Haeyoung1
Dokyung	Haeyoung1	Haeyoung2	Hun
Jinsang	Dokyung	Sukyung	Hun
Deogi	Haeyoung1	Dokyung	Jeongsuk
Sukyung	Jinsang	Haeyoung1	Jiya
Hun	Dokyung	Jinsang	Anna

Top3 person who the speaker talk about

Speaker	1st	2nd	3rd
Haeyoung1	Dokyung	Haeyoung2	Taejin
Jinsang	Haeyoung1	Taejin	Sukyung
Deogi	Haeyoung1	Dokyung	Taejin
Hun	Dokyung	Haeyoung2	Anna
Dokyung	Haeyoung1	Haeyoung2	Taejin
Haeyoung2	Haeyoung1	Dokyung	Chairman

(a)

(b)

Figure 4.6 (a) Top: Top-3 the person who the speaker frequently talks to, for each top 6 most spoken person. Bottom: Top-3 the person who the speaker frequently talks about, for each top 6 most spoken person. (b) The percentage of each person's utterance in the script.

4.2 Metrics

4.2.1 BLEU

BLEU(Bilingual Evaluation Understudy) is a method of measuring translation performance by comparing how similar machine translation results are to human translation results. The metric is based on n-grams. BLEU is not a perfect method, but it has several advantages. It can be used regardless of language, and the calculation speed is fast.

4.2.2 METEOR

METEOR(Metric for Evaluation of Translation with Explicit Ordering) is a metric for the evaluation of machine translation output. The metric is based on the harmonic mean of unigram precision and recall, with recall weighted higher than precision. It also has several features that are not found in other metrics, such as stemming and synonymy matching, along with the standard exact word matching

The metric was designed to fix some of the problems found in the more popular BLEU metric, also produce good correlation with human judgement at the sentence or segment level. This differs from the BLEU metric in that BLEU seeks correlation at the corpus level.

4.2.3 BERTScore

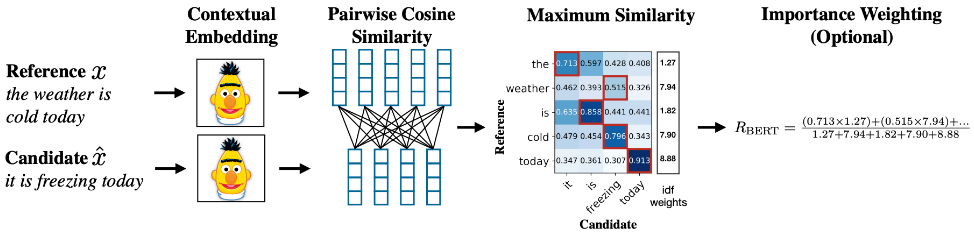
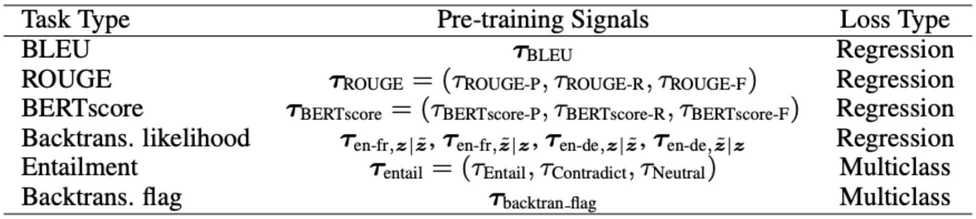


Figure 4.7 Illustration of the computation of the recall metric R_BERT. Given the reference x and candidate x^, it compute BERT embeddings and pairwise cosine similarity. The highlighted the greedy matching in red, and include the optional idf importance weighting

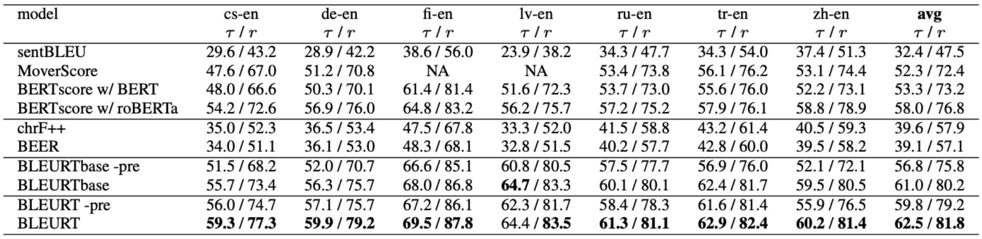
BERTScore is an automatic evaluation metric for text generation Analogously to common metrics, BERTSCORE computes a similarity score for each token in the candidate sentence with each token in the reference sentence. However, instead of exact matches, it computes token similarity using contextual embeddings. It evaluates using the outputs of 363 machine translation and image captioning systems. BERTSCORE correlates better with human judgments and provides stronger model selection performance than existing metrics. Finally, it uses an adversarial paraphrase detection task to show that BERTSCORE is more robust to challenging examples when compared to existing metrics.

4.3.4 BLEURT



Task Type	Pre-training Signals	Loss Type
BLEU	$\boldsymbol{\tau}_{\text{BLEU}}$	Regression
ROUGE	$\boldsymbol{\tau}_{\text{ROUGE}} = (\tau_{\text{ROUGE-P}}, \tau_{\text{ROUGE-R}}, \tau_{\text{ROUGE-F}})$	Regression
BERTscore	$\boldsymbol{\tau}_{\text{BERTscore}} = (\tau_{\text{BERTscore-P}}, \tau_{\text{BERTscore-R}}, \tau_{\text{BERTscore-F}})$	Regression
Backtrans. likelihood	$\boldsymbol{\tau}_{\text{en-fr},\boldsymbol{z} \mid \tilde{\boldsymbol{z}}}, \boldsymbol{\tau}_{\text{en-fr},\tilde{\boldsymbol{z}} \mid \boldsymbol{z}}, \boldsymbol{\tau}_{\text{en-de},\boldsymbol{z} \mid \tilde{\boldsymbol{z}}}, \boldsymbol{\tau}_{\text{en-de},\tilde{\boldsymbol{z}} \mid \boldsymbol{z}}$	Regression
Entailment	$\boldsymbol{\tau}_{\text{entail}} = (\tau_{\text{Entail}}, \tau_{\text{Contradict}}, \tau_{\text{Neutral}})$	Multiclass
Backtrans. flag	$\boldsymbol{\tau}_{\text{backtran\_flag}}$	Multiclass

Table 4.3 BLEURT's pre-training signals



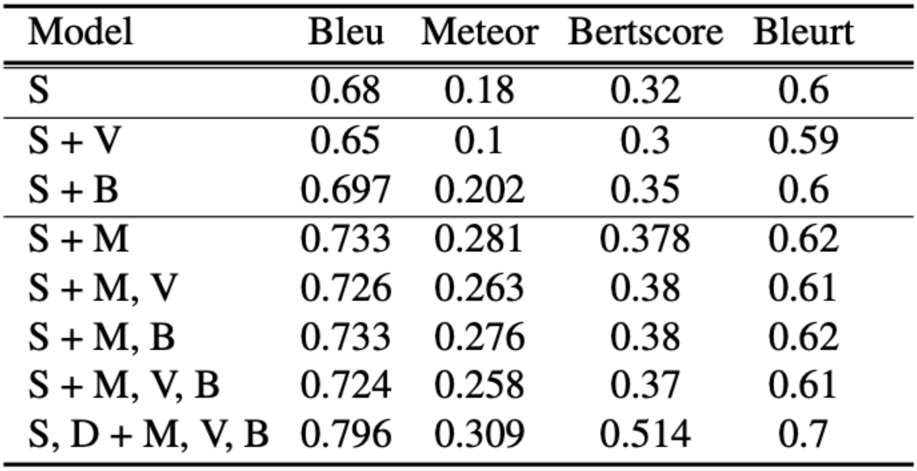
model	cs-en	de-en	fi-en	lv-en	ru-en	tr-en	zh-en	**avg**
	τ / r	τ / r	τ / r	τ / r	τ / r	τ / r	τ / r	τ / r
sentBLEU	29.6 / 43.2	28.9 / 42.2	38.6 / 56.0	23.9 / 38.2	34.3 / 47.7	34.3 / 54.0	37.4 / 51.3	32.4 / 47.5
MoverScore	47.6 / 67.0	51.2 / 70.8	NA	NA	53.4 / 73.8	56.1 / 76.2	53.1 / 74.4	52.3 / 72.4
BERTscore w/ BERT	48.0 / 66.6	50.3 / 70.1	61.4 / 81.4	51.6 / 72.3	53.7 / 73.0	55.6 / 76.0	52.2 / 73.1	53.3 / 73.2
BERTscore w/ roBERTa	54.2 / 72.6	56.9 / 76.0	64.8 / 83.2	56.2 / 75.7	57.2 / 75.2	57.9 / 76.1	58.8 / 78.9	58.0 / 76.8
chrF++	35.0 / 52.3	36.5 / 53.4	47.5 / 67.8	33.3 / 52.0	41.5 / 58.8	43.2 / 61.4	40.5 / 59.3	39.6 / 57.9
BEER	34.0 / 51.1	36.1 / 53.0	48.3 / 68.1	32.8 / 51.5	40.2 / 57.7	42.8 / 60.0	39.5 / 58.2	39.1 / 57.1
BLEURTbase -pre	51.5 / 68.2	52.0 / 70.7	66.6 / 85.1	60.8 / 80.5	57.5 / 77.7	56.9 / 76.0	52.1 / 72.1	56.8 / 75.8
BLEURTbase	55.7 / 73.4	56.3 / 75.7	68.0 / 86.8	**64.7** / 83.3	60.1 / 80.1	62.4 / 81.7	59.5 / 80.5	61.0 / 80.2
BLEURT -pre	56.0 / 74.7	57.1 / 75.7	67.2 / 86.1	62.3 / 81.7	58.4 / 78.3	61.6 / 81.4	55.9 / 76.5	59.8 / 79.2
BLEURT	**59.3 / 77.3**	**59.9 / 79.2**	**69.5 / 87.8**	64.4 / **83.5**	**61.3 / 81.1**	**62.9 / 82.4**	**60.2 / 81.4**	**62.5 / 81.8**

Table 4.4 Agreement with human ratings on the WMT17 Metrics Shared Task

The most popular choices for evaluating language generation model(e.g., BLEU and ROUGE) may correlate poorly with human judgments. BLEURT is a learned evaluation metric based on BERT that can model human judgments with a few thousand possibly biased training examples. A key aspect of this approach is a novel pre-training scheme that uses millions of synthetic examples to help the model generalize.

The evaluation is carried out using BLEU[14] based on n-gram, METEOR[2] considering recall as a traditional metric to evaluate the generated text. In addition, we evaluate the answers generated with a total of four metrics, including BERTScore[22] which is measured based on a similarity between each token embedding and BLEURT[16] which uses the pre-learned model as metric.

4.3 Quantitative Results



Model	Bleu	Meteor	Bertscore	Bleurt
S	0.68	0.18	0.32	0.6
S + V	0.65	0.1	0.3	0.59
S + B	0.697	0.202	0.35	0.6
S + M	0.733	0.281	0.378	0.62
S + M, V	0.726	0.263	0.38	0.61
S + M, B	0.733	0.276	0.38	0.62
S + M, V, B	0.724	0.258	0.37	0.61
S, D + M, V, B	0.796	0.309	0.514	0.7

Table 4.4 Quantitative experimental results for the DramaQA validation set. S stand for subtitle, V stands for video features extracted from I3D, B stands for bounding box features extracted from ResNet, and M stands for visual metadata composed of person, emotion, and behavior, D stands for description for scene.

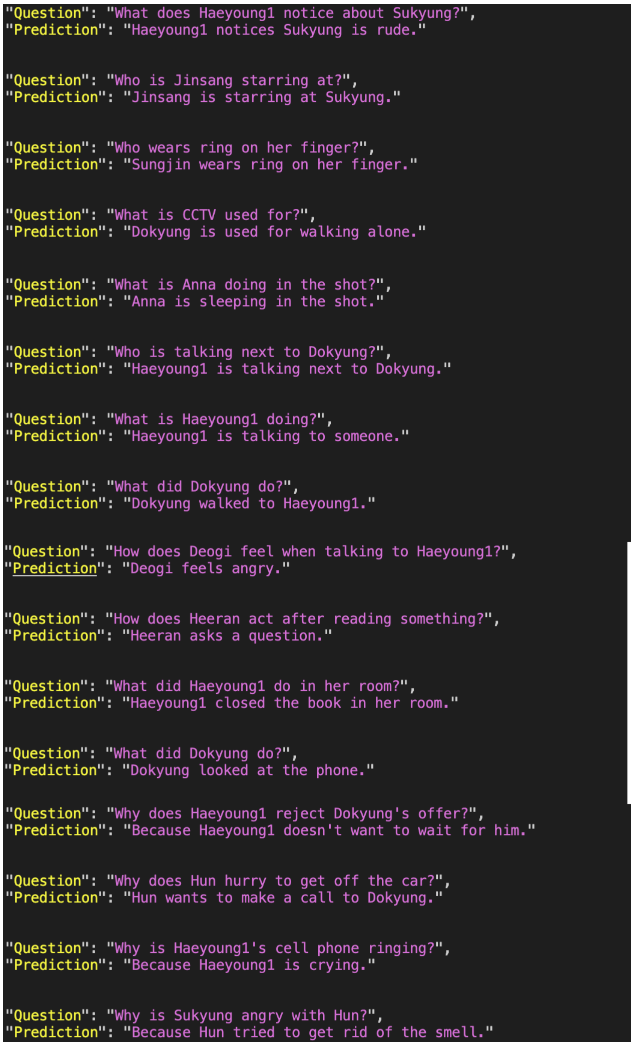
Table 3 shows metadata plays a major role in improving performance. Our model is based on GPT2, so there is language bias. It helps improve performance with language metadata.

The information in bounding box features also helps answer questions by looking at S / B + S. However, comparing M + S / B, M + S did not improve performance.

Video information lowers performance. For reasons, a transformer-based model is a model with large language bias, and the entire video that is irrelevant to the question works even worse than bounding box features.

Finally, it can be seen that the descriptive information on Scene greatly affects the performance of the model. Since it is a language-based model, meaningful text data can greatly improve the performance of the model. Video capping can be used to replace the limited situation in which the description is directly applied.

4.4 Qualitative Results



```
"Question": "What does Haeyoung1 notice about Sukyung?",
"Prediction": "Haeyoung1 notices Sukyung is rude."

"Question": "Who is Jinsang starring at?",
"Prediction": "Jinsang is starring at Sukyung."

"Question": "Who wears ring on her finger?",
"Prediction": "Sungjin wears ring on her finger."

"Question": "What is CCTV used for?",
"Prediction": "Dokyung is used for walking alone."

"Question": "What is Anna doing in the shot?",
"Prediction": "Anna is sleeping in the shot."

"Question": "Who is talking next to Dokyung?",
"Prediction": "Haeyoung1 is talking next to Dokyung."

"Question": "What is Haeyoung1 doing?",
"Prediction": "Haeyoung1 is talking to someone."

"Question": "What did Dokyung do?",
"Prediction": "Dokyung walked to Haeyoung1."

"Question": "How does Deogi feel when talking to Haeyoung1?",
"Prediction": "Deogi feels angry."

"Question": "How does Heeran act after reading something?",
"Prediction": "Heeran asks a question."

"Question": "What did Haeyoung1 do in her room?",
"Prediction": "Haeyoung1 closed the book in her room."

"Question": "What did Dokyung do?",
"Prediction": "Dokyung looked at the phone."

"Question": "Why does Haeyoung1 reject Dokyung's offer?",
"Prediction": "Because Haeyoung1 doesn't want to wait for him."

"Question": "Why does Hun hurry to get off the car?",
"Prediction": "Hun wants to make a call to Dokyung."

"Question": "Why is Haeyoung1's cell phone ringing?",
"Prediction": "Because Haeyoung1 is crying."

"Question": "Why is Sukyung angry with Hun?",
"Prediction": "Because Hun tried to get rid of the smell."
```

Figure 4.8 Qualitative results. question prediction pairs for levels 1, 2, 3, and 4 are shown.

Chapter 5

Conclusion

In this paper, we challenge the existing multiple-choice video question answer by converting it into an open-ended form. We construct the model in the form of a multimodal transformer by adding video and metadata from video to the existing pre-trained language model. Ablation studies using the DramaQA dataset showed that video metadata helped performance.

For future work, we plan to use the dense caption features in the video space transferred into the language space to circumvent the language bias problem. As a result of using description data in DramaQA Dataset for verification, it showed remarkable performance improvement.

In addition, performance can be improved by using language models such as chatgpt, Galactica, and GPT3 that have recently been released.

Reference

[1] Huda Alamri, Vincent Cartillier, Abhishek Das, Jue Wang, Anoop Cherian, Irfan Essa, Dhruv Batra, Tim K. Marks, Chiori Hori, Peter Anderson, Stefan Lee, and Devi Parikh. Audio-visual scene-aware dialog, 2019. 1, 2

[2] Satanjeev Banerjee and Alon Lavie. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pages 65–72, 2005. 4

[3] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6299–6308, 2017. 2

[4] Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. Uniter: Universal image-text representation learning. In *ECCV*, 2020. 1

[5] Seongho Choi, Kyoung-Woon On, Yu-Jung Heo, Ahjeong Seo, Youwon Jang, Seungchan Lee, Minsu Lee, and Byoung Tak Zhang. Dramaqa: Character-centered video story understanding with hierarchical qa, 2020. 1

[6] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 2

[7] Shawn Hershey, Sourish Chaudhuri, Daniel P. W. Ellis, Jort F. Gemmeke, Aren Jansen, Channing Moore, Manoj Plakal, Devin Platt, Rif A. Saurous, Bryan Seybold, Malcolm Slaney, Ron Weiss, and Kevin Wilson. Cnn architectures for large-scale audio classification. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2017. 2

[8] Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. The curious case of neural text degeneration. *arXiv preprint arXiv:1904.09751*, 2019. 3

[9] Yunseok Jang, Yale Song, Youngjae Yu, Youngjin Kim, and Gunhee Kim. Tgif-qa: Toward spatio-temporal reasoning in visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2758–2766, 2017. 1

[10] Kyung-Min Kim, Min-Oh Heo, Seong-Ho Choi, and Byoung-Tak Zhang. Deepstory: Video story qa by deep embedded memory networks, 2017. 1, 2

[11] Jie Lei, Licheng Yu, Mohit Bansal, and Tamara L. Berg. Tvqa: Localized, compositional video question answering, 2019. 1, 2

[12] Zekang Li, Zongjia Li, Jinchao Zhang, Yang Feng, Cheng Niu, and Jie Zhou. Bridging text and video: A universal multimodal transformer for video-audio scene-aware dialog, 2020. 2

[13] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 3

[14] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318, 2002. 4

[15] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019. 2

[16] Thibault Sellam, Dipanjan Das, and Ankur P Parikh. Bleurt: Learning robust metrics for text generation. *arXiv preprint arXiv:2004.04696*, 2020. 4

[17] Makarand Tapaswi, Yukun Zhu, Rainer Stiefelhagen, Antonio Torralba, Raquel Urtasun, and Sanja Fidler. Movieqa: Understanding stories in movies through question answering, 2016. 1, 2

[18] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017. 2

[19] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, undefinedukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 6000–6010, Red Hook, NY, USA, 2017. Curran Associates Inc. 1

[20] Hongyang Xue, Zhou Zhao, and Deng Cai. Unifying the video and question attentions for open-ended video question answering. *IEEE Transactions on Image Processing*, 26(12):5656–5666, 2017. 2

[21] Zekun Yang, Noa Garcia, Chenhui Chu, Mayu Otani, Yuta Nakashima, and Haruo Takemura. Bert representations for video question answering. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1556–1565, 2020. 2

[22] Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*, 2019. 4

[23] Zhou Zhao, Zhu Zhang, Shuwen Xiao, Zhou Yu, Jun Yu, Deng Cai, Fei Wu, and Yueting Zhuang. Open-ended long form video question answering via adaptive hierarchical reinforced networks. In *IJCAI*, volume 3, page 4, 2018. 2

국문 초록

비디오 질의응답은 최근 멀티 모달 비디오 연구자들로부터 많은 관심을 받고 있다. 대부분의 비디오 질의응답 데이터셋은 객관식 질의응답의 형식으로 제공되고 있다. 그러나 객관식 질의응답 태스크는 답을 추론하지 않고, 답안 후보군들을 비교해서 더 나은 선택지를 찾는 방식을 택한다. 이러한 방식은 선택지에 제한되어 비디오와 질문 간의 상세한 상호작용을 파악하기 어렵다. 그에 반해 주관식의 경우 자유로운 답변 생성을 통해 모델이 비디오와 질문 사이의 복잡한 관계를 이해하기에 용이하다. 뿐만 아니라 실용적인 측면에서 인간과의 상호작용을 위해서는 답안 후보군을 제공하는 방식보다 주관식으로의 상호작용이 더 용이하다. 본 논문에서는 기존의 객관식 질의응답 문제를 주관식 질의응답으로 바꿔서 앞서 말한 문제들을 해결하고자 한다. 주관식 질의 응답 문제를 해결하기 위해 미리 학습된 GPT2 model을 활용한다. 비디오의 내용을 이해하기 위해서는 등장인물, 사건에 대한 정보들이 필요하다. 이를 위해 비디오 입력, 자막, 메타데이터, 디스크립션 등의 정보를 활용해 파인 튜닝한다. 본 연구에서는 기존의 DramaQA 데이터셋을 주관식 질의응답이 가능한 형태로 변형해 수행되었다. 비디오 메타데이터, 디스크립션을 활용해 주관식 질의응답 문제에 높은 성능을 보였다.

주요어 : 비디오, 비디오 질의응답, 주관식, transformer, visual metadata, Language Model

학 번 : 2021-27378