

# VIDEO QUESTION ANSWERING USING CLIP-GUIDED VISUAL-TEXT ATTENTION

Shuhong Ye<sup>\*§</sup>, Weikai Kong<sup>\*§</sup>, Chenglin Yao<sup>\*</sup>, Jianfeng Ren<sup>\*†</sup>, Xudong Jiang<sup>‡</sup>

<sup>\*</sup>School of Computer Science, University of Nottingham Ningbo China

<sup>†</sup>Nottingham Ningbo China Beacons of Excellence Research and Innovation Institute,  
University of Nottingham Ningbo China, China

<sup>‡</sup>School of Electrical & Electronic Engineering, Nanyang Technological University

## ABSTRACT

Cross-modal learning of video and text plays a key role in Video Question Answering (VideoQA). In this paper, we propose a visual-text attention mechanism to utilize the Contrastive Language-Image Pre-training (CLIP) trained on lots of general domain language-image pairs to guide the cross-modal learning for VideoQA. Specifically, we first extract video features using a TimeSformer and text features using a BERT from the target application domain, and utilize CLIP to extract a pair of visual-text features from the general-knowledge domain through the domain-specific learning. We then propose a Cross-domain Learning to extract the attention information between visual and linguistic features across the target domain and general domain. The set of CLIP-guided visual-text features are integrated to predict the answer. The proposed method is evaluated on MSVD-QA and MSRVT-QA datasets, and outperforms state-of-the-art methods.

**Index Terms**— Video Question Answering, CLIP, Cross-modal Learning, Cross-domain Learning

## 1. INTRODUCTION

Video Question Answering (VideoQA) has become increasingly popular in vision-language navigation [1], multimedia recommendation [2], and communication systems [3]. The target is to correctly answer questions about a video, which requires a deep understanding of video scenes, question semantics, and fine-grained vision-language alignment. Many methods have been developed for cross-modality learning [4–7]. But due to limited training data, some linguistic concepts in the answer space have no corresponding video samples, resulting in the lack of linguistic supervision. Such a problem hinders accurate pairing between linguistic features and corresponding visual features, and hence limits the cross-modality learning ability of the existing models [8–14].

There are two main directions for improving VideoQA models. 1) Exploit deeper correlations from annotated video-text pairs using recurrent neural network [9, 10], graph neural network [11, 14, 15], conditional relation network [13], or attention-based models [8, 12]. 2) Overcome the challenge of insufficient linguistic supervision by importing general domain knowledge from large-scale pre-trained vision-language models, *e.g.*, HERO [16], ClipBert [6], and JustAsk [17]. Benefiting from the additional general domain knowledge that can describe unseen answers in the target application domain, pre-trained models significantly boost the performance of downstream VideoQA tasks and achieve state-of-the-art results on many VideoQA datasets [4, 8, 18].

It should be noted there may be a knowledge discrepancy between the target domain and the general domain. Failing to bridge the discrepancy may bring conflicts in the cross-modal knowledge representation. To address this problem, we propose a two-stage cross-domain cross-modal learning framework under the guidance of CLIP model [19]. CLIP is pre-trained from large-scale image-text pairs so that more unseen answers can be described. It provides a bidirectional transform of features from the salient contents in video frames and text captions. In the first stage, vision-language features are extracted in a domain-specific way. In the target domain, the video features are extracted via a TimeSformer [20] and the question and answer texts are encoded using transformers [21, 22]. To incorporate the general domain knowledge from the large-scale pre-trained CLIP, the key video frames are extracted as the salient contents and fed into the CLIP to generate the CLIP-guided visual features, and the question and answers are fed into the CLIP to generate the CLIP-guided linguistic features. In the second stage of cross-domain learning, to bridge the knowledge discrepancy, four CLIP-guided visual-text encoders are designed to exploit the cross-modal cross-domain attention information. Finally, the four sets of visual-text features are fused to predict the answer.

Our contributions are three-fold: 1) The proposed method effectively extracts the visual and linguistic features from the general domain knowledge using the pre-trained CLIP and from the target domain using TimeSformer and transformer.

<sup>§</sup> The authors contributed equally.

This work was supported in part by the National Natural Science Foundation of China under Grant 72071116, and in part by the Ningbo Municipal Bureau Science and Technology under Grants 2019B10026 and 2022Z173.

2) The proposed CLIP-guided visual-text attention mechanism effectively integrates the general domain knowledge into the target domain for cross-modal cross-domain learning in VideoQA. 3) Experimental results on two large benchmark datasets demonstrate that the proposed method significantly outperforms state-of-the-art VideoQA models.

## 2. PROPOSED METHOD

### 2.1. Overview of Proposed Method

The block diagram of the proposed CLIP-guided Cross-domain Video Question Answering (CCVQA) model is shown in Fig 1. It consists of two main modules. 1) Domain-specific Learning. The vision and language features are first extracted in a domain-specific way. More specifically, a TimeSformer [20] is utilized to extract the visual features from the video sequences in the target VideoQA domain. To incorporate the general domain knowledge, the key frames of the video are first selected and fed into the image stream of CLIP [19] to extract visual features that are compatible with language supervision. Similarly, the transformer [21, 22] and the text stream of CLIP [19] take question sentences and prompts as the input to extract text features in the target domain and general domain, respectively. 2) Cross-domain Learning. After extracting the visual and language features separately in the target domain or in the general domain, these four sets of features are integrated through cross-domain learning. Four CLIP-guided visual-text encoders are designed for cross-domain cross-modal learning through the attention mechanism. Finally, the four sets of extracted features are fused through a multi-layer perceptron (MLP), and the answer decoder [10, 13] is adopted to derive the correct answer.

### 2.2. Domain-specific Learning

Given a video with a question to answer, the vision feature extraction, language feature extraction, and vision-language alignment are three critical steps for answering the question [8–14], while lack of general domain knowledge often makes the open-ended VideoQA very challenging [6, 23, 24]. To address this challenge, CLIP [19] is integrated into the proposed method to incorporate the general domain knowledge. The proposed CCVQA model consists of two visual encoders and two language encoders.

In the target domain, a 12-layer TimeSformer [20] is used to encode the spatial-temporal information embedded in each video frame. 16 frames are randomly sampled from a video to preserve as much information as possible while keeping a low computational load. The TimeSformer partitions each frame into  $N_v$  patches and generates patch tokens using a linear projection layer. After learnable positional embedding is added, the tokens are fed into the attention blocks to perform self-attention across the temporal and spatial dimensions. A tem-

poral fusion layer aggregates the outputs from the attention blocks along the temporal dimension to obtain the sequenced features  $\mathbf{H}_v = [v_{cls}, v_1, \dots, v_{N_v}]$ , with  $v_i \in \mathbb{R}^d$ .  $d$  is the dimension of video features.  $v_{cls}$  is the classification token.

In the target domain, a 6-layer BERT-base model [25] is used to encode the language features  $\mathbf{H}_q$ . Given the input question of  $N_q$  tokens, the BERT sequentially performs self-attention on the input and outputs an embedding sequence  $\mathbf{H}_q = [q_{cls}, q_1, \dots, q_{N_q}] \in \mathbb{R}^{N_q \times d}$ ,  $q_i \in \mathbb{R}^d$  and  $q_{cls}$  is the [CLS] token.  $d$  is the dimension of question features, the same as that of video features. Learnable positional embeddings are added to the text tokens, similar to the video encoder.

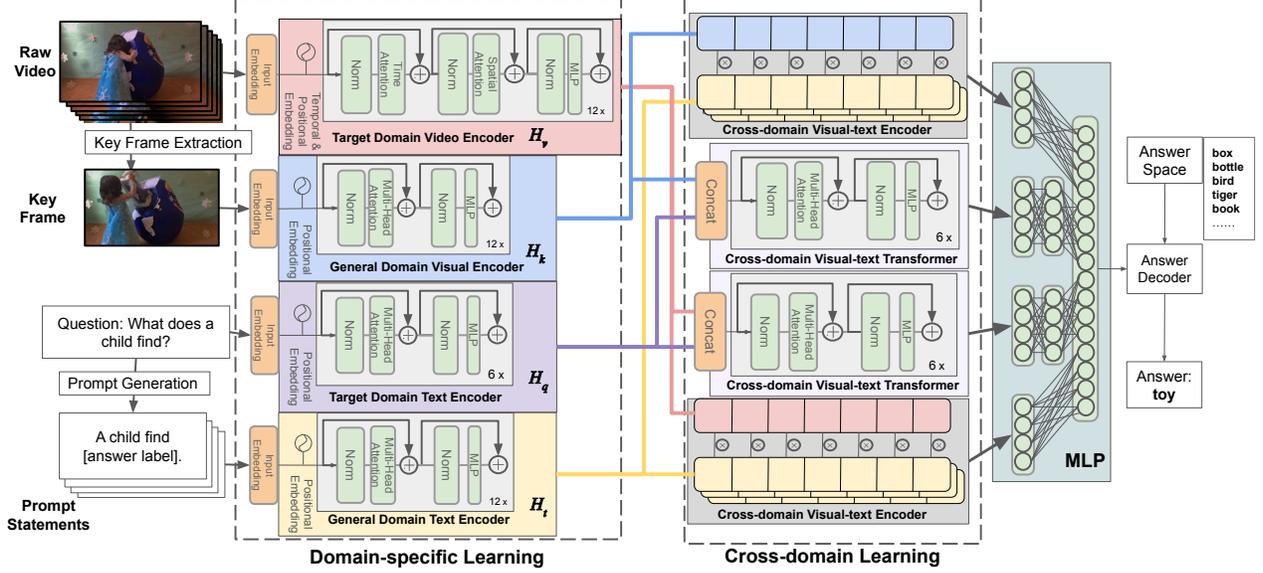
To incorporate general domain knowledge, a CLIP-guided visual encoder is designed. As it is time-consuming to encode the whole video sequence using CLIP, only key frames are selected according to the color histogram contrast<sup>1</sup> and encoded using the CLIP ViT-B/32 model [22]. The CLIP is pre-trained on 400 million image-text pairs collected from the Internet, to map visual-text features into a joint embedding space, which can catch the similarity between encoded image features and linguistic features, and serve as additional language supervision to guide the subsequent text-image alignment. The CLIP firstly transforms a key frame into  $N_k$  patches using linear projections. After adding the positional embedding, self-attentions are performed on the patches together with the [CLS] token to output the sequence features of the key frame  $\mathbf{H}_k = [k_{cls}, k_1, \dots, k_{N_k}]$ ,  $k_i \in \mathbb{R}^d$ .

The CLIP-guided text encoder is designed to extract the linguistic features using the general domain knowledge. The given question is used to generate the prompt, with the key information extracted from the question and the possible answer from the given answer space. Note that we are dealing with open-ended QA and hence there is no answer option but a very large space of  $C$  possible answers. Similar to the CLIP-guided visual encoder, the CLIP language encoder tokenizes the prompt sentences, linearly projects them into the embedding space, and adds position embeddings with a layer normalization operation. The self-attention is then performed on the sentence embedding and produces a batch of feature vectors  $\mathbf{H}_t \in \mathbb{R}^{C \times N_t \times w}$ , where  $N_t$  is the sequence length for each of the encoded prompt features  $[t_1, \dots, t_{N_t}]$ , with  $t_i \in \mathbb{R}^w$ ,  $w$  is the prompt feature dimensionality.

### 2.3. Cross-domain Learning

After deriving the visual features  $\mathbf{H}_v$ ,  $\mathbf{H}_k$  and linguistic features  $\mathbf{H}_q$ ,  $\mathbf{H}_t$  in a domain-specific way, four cross-domain cross-modal visual-text encoders are designed to capture the attentional information among these features. A 6-layer shared-weight transformer  $\mathcal{T}$  is designed to model the attentional information between  $\mathbf{H}_q$  and  $\mathbf{H}_v$ , and between  $\mathbf{H}_q$  and  $\mathbf{H}_k$ . It directly takes the concatenated multi-modal features as the input, performs self-attention on paired visual features

<sup>1</sup>Code is available at <https://github.com/keplerlab/katna>



**Fig. 1.** Overview of the proposed CCVQA. It consists of two main modules. 1) Domain-specific Learning, including a video encoder to extract spatial-temporal features  $H_v$ , a text encoder to encode question descriptions  $H_q$  from the target domain, a CLIP-guided video key frame encoder to encode object representations  $H_k$  with language supervision, and a CLIP-guided candidate answer encoder to generate linguistic features  $H_t$  with vision supervision in the general domain. 2) Cross-domain Learning, including four sets of CLIP-guided visual-text encoders to model cross-domain cross-modal feature interaction through the attention mechanism. Finally, the features are fused using an MLP and fed to a decoder to predict the answer.

and linguistic features, and produces the class tokens with a MLP as formatted in Eqn. (1) and (2).

$$H_{qv} = W_{qv} \mathcal{T}(H_q; H_v) + b_{qv}, \quad (1)$$

$$H_{qk} = W_{qk} \mathcal{T}(H_q; H_k) + b_{qk}, \quad (2)$$

where the class tokens  $H_{qv}, H_{qk} \in \mathbb{R}^a$ .  $W_{qv}, b_{qv}, W_{qk}$ , and  $b_{qk}$  are trainable parameters.

The size of CLIP encoded prompts  $H_t \in \mathbb{R}^{a \times N_k \times w}$  is much larger than that of the question features  $H_q \in \mathbb{R}^{N_q \times d}$ . The concatenation of  $H_t$  and  $H_v$  or  $H_k$  will result in large sequences and computing such sequences in transformer can take up to some GPU years [19]. Therefore, to simplify the extraction of attentional information using self-attention via a transformer, we perform dot product for the interactions between the prompt text features  $H_t$  and the visual features  $H_v$  or  $H_k$ . The [CLS] tokens of  $H_t$ ,  $H_v$  and  $H_k$  are linearly projected to vectors with the size of  $\mathbb{R}^{a \times w}$ ,  $\mathbb{R}^w$  and  $\mathbb{R}^w$ , respectively. Then the two vectors are projected and multiplied to produce another two class tokens,

$$H_{tv} = P(H_t) \odot P(H_v), \quad (3)$$

$$H_{tk} = P(H_t) \odot P(H_k), \quad (4)$$

where  $H_{tv}, H_{tk} \in \mathbb{R}^a$ ,  $\odot$  denotes the dot product, and  $P(\cdot)$  denotes the projection.

A weighted multi-head fusion is then utilized to integrate the four tokens  $H_{qv}, H_{qk}, H_{tv}$ , and  $H_{tk}$ . For each token, a

linear layer is utilized for feature alignment so that each item on the token represents the confidence level for one class from the answer space. The final fused cross-domain cross-modal features  $H$  are derived as in Eqn. (5),

$$H = W_1 H_{qv} + W_2 H_{qk} + W_3 H_{tv} + W_4 H_{tk} + b \quad (5)$$

where  $W_1, W_2, W_3, W_4$ , and  $b$  are trainable fusion weights.

### 3. EXPERIMENTAL RESULTS

#### 3.1. Experimental Settings

The proposed CCVQA is evaluated on two public datasets for open-ended VideoQA: MSVD-QA [8] and MSRVT-QA [8], which are generated using videos from the original datasets, MSVD [26] and MSRVT [27], respectively, combined with auto-generated question answer annotation pairs. MSVD-QA contains 1.9K videos and 50k question-answer pairs in total and MSRVT-QA has 10K videos and 243k question-answer pairs in total. The questions are categorized into five types: "what", "who", "when", "where", and "how" based on the starting words of the questions. The answers are all one-word nouns or verbs describing concrete objects or abstract concepts in the video. The standard dataset partition is followed [8]. On the MSVD-QA dataset, 61% of the data are used for training, 13% for validation, and 26% for testing. On the MSRVT-QA dataset, the train-validation-test data split

is 65%, 5%, and 30%. As for open-ended question answers, 2,423 answer candidates are used for MSVD-QA and 1,500 for MSRVTT-QA. The proposed method is compared with state-of-the-art models. AMU [8], Co-mem [9], HME [10], HGA [11], SSML [28], QUEST [12], HCRN [13], STN [29], DualVGR [15], and HQGA [14] are selected as representative cross-modal learning methods. ClipBERT [6], CoMVT [23], SSRea [24], and VQA-T [17] are selected as typical pre-train model-enhanced methods.

The CLIP ViT-B/32 architecture with pre-trained weights is obtained from the OpenAI’s official release<sup>2</sup>. All video frames are resized to  $224 \times 224$ . On language feature encoding, the question is converted to the statement form at a size of 1,500 sentences on MSRVTT-QA dataset and 2,423 sentences on MSVD-QA dataset which are the same as the number of answer candidates. All experiments were performed on a single NVIDIA V100 GPU. The initial learning rate is set to  $5 \times 10^{-5}$  and linearly decayed in the following epochs. The batch size is 24 for the training processes on both datasets. The AdamW optimizer with a weight decay of  $1 \times 10^{-3}$  is employed. The model converges within 15 training epochs on MSVD-QA and 10 epochs on MSRVTT-QA, consuming 120 GPU hours and 400 GPU hours respectively.

### 3.2. Comparisons to State-of-the-art Models

Methods	MSVD-QA	MSRVTT-QA
AMU [8]	32.0	32.5
Co-mem [9]	31.7	32.0
HME [10]	33.7	33.0
HGA [11]	34.7	35.5
SSML [28]	35.13	35.06
QUEST [12]	36.1	34.6
HCRN [13]	36.1	35.6
TSN [29]	36.7	35.4
DualVGR [15]	39.03	35.52
HQGA [14]	41.2	38.6
ClipBERT [6]	-	37.4
CoMVT [23]	42.6	39.5
SSRea [24]	45.5	41.6
<b>Ours</b>	<b>46.6</b>	<b>42.4</b>

**Table 1.** Comparisons with state-of-the-art methods on MSRVTT-QA and MSVD-QA datasets in top-1 accuracy (%).

The comparisons with state-of-the-art methods are listed in Table 1. The results confirm that our model outperforms all the compared methods by a considerable margin. The proposed method improves the top-1 accuracy from 45.5% achieved by SSRea [24] to 46.6% on the MSVD-QA dataset and from 41.6% to 42.4% on the MSRVTT-QA dataset, re-

spectively. Compared with the VideoQA methods such as HQGA [14] and DualVGR [15] that do not use pre-trained models, the proposed model can significantly outperform them by nearly 5.4% in accuracy on the MSVD-QA dataset and 3.8% on the MSRVTT-QA dataset, which validates the effectiveness of using additional general domain knowledge. Compared to other pre-trained models such as SSRea [24] and CoMVT [23], the proposed model further boosts the accuracy by around 1.1% on the MSVD-QA dataset and 0.8% on the MSRVTT-QA dataset, which demonstrates the effectiveness of our visual and linguistic features encoders and CLIP-guided visual-text attention mechanism.

### 3.3. Ablation Studies

To show the performance gain brought about by each contribution, an ablation study is conducted. The proposed CCVQA is compared with two baselines: 1) **CCVQA w/o CLIP**, where the linguistic and visual clues from CLIP are not used, and 2) **CCVQA w/o Cross-domain Learning**, where only visual-text learning within the target/general domain is applied. As shown in Table 2, CCVQA achieves a significant performance gain of 1.9% and 1.1% over **CCVQA w/o CLIP** on the MSVD-QA and MSRVTT-QA datasets, respectively, which demonstrates the effectiveness of our CLIP-guided design. CCVQA also achieves a performance gain of 0.7% and 0.3% through the proposed Cross-domain Learning on the MSVD-QA and MSRVTT-QA datasets, respectively.

Method	MSVD-QA	MSRVTT-QA
CCVQA w/o CLIP	44.7	41.3
CCVQA w/o Cross-domain Learning	45.9	42.1
<b>CCVQA</b>	<b>46.6</b>	<b>42.4</b>

**Table 2.** Ablation studies on the MSRVTT-QA and MSVD-QA datasets in terms of top-1 accuracy(%).

## 4. CONCLUSION

To tackle the challenge of insufficient language supervision in VideoQA, a CLIP-guided cross-domain video-text encoder is proposed to transform CLIP’s general domain knowledge into the application domain. Specifically, the pre-trained CLIP encodes key video frames and prompt texts using general domain knowledge. Together with video features extracted by the TimeSformer and question text features encoded by the BERT in the application domain, the attentional information across visual and text features are extracted using dot product and shared-weight transformer through cross-domain learning. Finally, the answer is decoded from the answer space. The proposed CCVQA is evaluated on two open-ended VideoQA datasets, MSVD-QA and MSRVTT-QA, which demonstrates a consistent and significant performance gain over the state-of-the-art models.

<sup>2</sup>CLIP model is available at <https://github.com/openai/CLIP>

## References

- [1] P. Anderson, Q. Wu, D. Teney, J. Bruce, M. Johnson, N. Sünderhauf, I. Reid, S. Gould, and A. Van Den Hengel, "Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments," in *CVPR*, 2018, pp. 3674–3683. 1
- [2] J. Chen, H. Zhang, X. He, L. Nie, W. Liu, and T. Chua, "Attentive collaborative filtering: Multimedia recommendation with item-and component-level attention," in *SIGIR*, 2017, pp. 335–344. 1
- [3] M. Bica, K. Huang, V. Koivunen, and U. Mitra, "Mutual information based radar waveform design for joint radar and cellular communication systems," in *ICASSP*, 2016, pp. 3671–3675. 1
- [4] J. Lei, M. Yu, L. Bansal, and T. Berg, "Tvqa: Localized, compositional video question answering," in *EMNLP*, 2018, pp. 1369–1379. 1
- [5] J. Lei, L. Yu, T. Berg, and M. Bansal, "Tvqa+: Spatio-temporal grounding for video question answering," in *ACL*, 2020, pp. 8211–8225.
- [6] J. Lei, L. Li, L. Zhou, Z. Gan, T. Berg, M. Bansal, and J. Liu, "Less is more: Clipbert for video-and-language learning via sparse sampling," in *CVPR*, 2021, pp. 7331–7341. 1, 2, 4
- [7] A. Guzhov, F. Raue, J. Hees, and A. Dengel, "Audioclip: Extending clip to image, text and audio," in *ICASSP*, 2022, pp. 976–980. 1
- [8] D. Xu, Z. Zhao, J. Xiao, F. Wu, H. Zhang, X. He, and Y. Zhuang, "Video question answering via gradually refined attention over appearance and motion," in *ACM MM*, 2017, pp. 1645–1653. 1, 2, 3, 4
- [9] J. Gao, R. Ge, K. Chen, and R. Nevatia, "Motion-appearance co-memory networks for video question answering," in *CVPR*, 2018, pp. 6576–6585. 1, 4
- [10] C. Fan, X. Zhang, S. Zhang, W. Wang, C. Zhang, and H. Huang, "Heterogeneous memory enhanced multimodal attention model for video question answering," in *CVPR*, 2019, pp. 1999–2007. 1, 2, 4
- [11] P. Jiang and Y. Han, "Reasoning with heterogeneous graph alignment for video question answering," in *AAAI*, 2020, vol. 34, pp. 11109–11116. 1, 4
- [12] J. Jiang, Z. Chen, H. Lin, X. Zhao, and Y. Gao, "Divide and conquer: Question-guided spatio-temporal contextual attention for video question answering," in *AAAI*, 2020, vol. 34, pp. 11101–11108. 1, 4
- [13] T. Le, V. Le, S. Venkatesh, and T. Tran, "Hierarchical conditional relation networks for video question answering," in *CVPR*, 2020, pp. 9972–9981. 1, 2, 4
- [14] J. Xiao, A. Yao, Z. Liu, Y. Li, W. Ji, and T. Chua, "Video as conditional graph hierarchy for multi-granular question answering," in *AAAI*, 2022, pp. 2804–2812. 1, 2, 4
- [15] J. Wang, B. Bao, and C. Xu, "Dualvgr: A dual-visual graph reasoning unit for video question answering," *IEEE Transactions on Multimedia*, vol. 24, pp. 3369–3380, 2021. 1, 4
- [16] L. Li, Y. Chen, Y. Cheng, Z. Gan, L. Yu, and J. Liu, "Hero: Hierarchical encoder for video+ language omni-representation pre-training," in *EMNLP*, 2020, pp. 2046–2065. 1
- [17] A. Yang, A. Miech, J. Sivic, I. Laptev, and C. Schmid, "Just ask: Learning to answer questions from millions of narrated videos," in *CVPR*, 2021, pp. 1686–1697. 1, 4
- [18] Y. Yu, J. Kim, and G. Kim, "A joint sequence fusion model for video question answering and retrieval," in *ECCV*, 2018, pp. 471–487. 1
- [19] A. Radford, J. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, et al., "Learning transferable visual models from natural language supervision," in *ICML*, 2021, pp. 8748–8763. 1, 2, 3
- [20] C. Bertasius, H. Wang, and L. Torresani, "Is space-time attention all you need for video understanding?," in *ICML*, 813–824, vol. 139, p. 4. 1, 2
- [21] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. Gomez, Lu. Kaiser, and I. Polosukhin, "Attention is all you need," *NeurIPS*, vol. 30, 2017. 1, 2
- [22] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," in *ICLR*, 2021. 1, 2
- [23] P. Seo, A. Nagrani, and C. Schmid, "Look before you speak: Visually contextualized utterances," in *CVPR*, 2021, pp. 16877–16887. 2, 4
- [24] W. Yu, H. Zheng, M. Li, L. Ji, L. Wu, N. Xiao, and N. Duan, "Learning from inside: Self-driven siamese sampling and reasoning for video question answering," *NeurIPS*, vol. 34, pp. 26462–26474, 2021. 2, 4
- [25] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018. 2
- [26] D. Chen and W. Dolan, "Collecting highly parallel data for paraphrase evaluation," in *ACL*, 2011, pp. 190–200. 3
- [27] J. Xu, T. Mei, T. Yao, and Y. Rui, "Msr-vtt: A large video description dataset for bridging video and language," in *CVPR*, 2016, pp. 5288–5296. 3
- [28] E. Amrani, R. Ben-Ari, D. Rotman, and A. Bronstein, "Noise estimation using density estimation for self-supervised multimodal learning," in *AAAI*, 2021, vol. 35, pp. 6644–6652. 4
- [29] T. Yang, Z. Zha, H. Xie, M. Wang, and H. Zhang, "Question-aware tube-switch network for video question answering," in *ACM MM*, 2019, pp. 1184–1192. 4