Tem-adapter: Adapting Image-Text Pretraining for Video Question Answer

Guangyi Chen[1,2*], Xiao Liu[5*], Guangrun Wang[4], Kun Zhang[1,2], Philip H.S. Torr[4],
Xiao-Ping Zhang[3,6], Yansong Tang[3†]
[1]Carnegie Mellon University, Pittsburgh PA, USA
[2]Mohamed bin Zayed University of Artificial Intelligence, Abu Dhabi, UAE
[3]Shenzhen International Graduate School, Tsinghua University, China
[4]University of Oxford, UK
[5]Eindhoven University of Technology, NL
[6]Toronto Metropolitan University, Canada

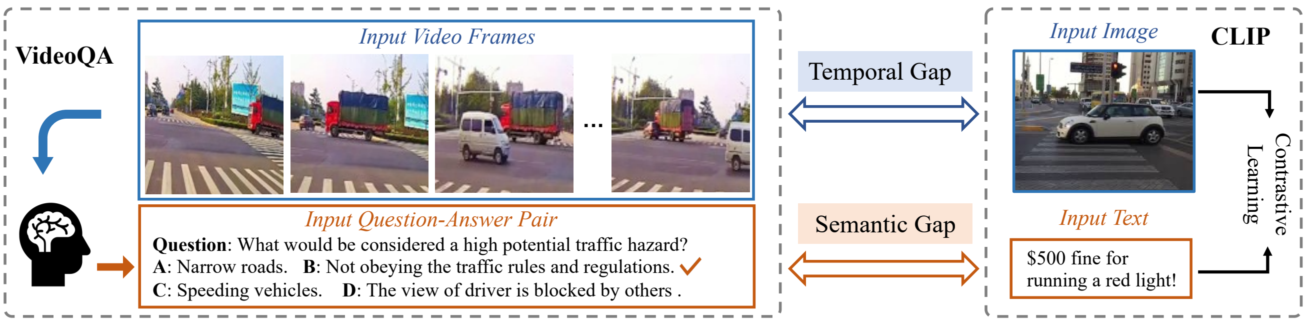

Figure 1: Illustration of the domain gaps between the pre-trained CLIP and the downstream VideoQA. CLIP is trained to align visual and textual domains, while VideoQA requires understanding the temporal dynamics and complex semantics of videos.

Abstract

Video-language pre-trained models have shown remarkable success in guiding video question-answering (VideoQA) tasks. However, due to the length of video sequences, training large-scale video-based models incurs considerably higher costs than training image-based ones. This motivates us to leverage the knowledge from image-based pretraining, despite the obvious gaps between image and video domains. To bridge these gaps, in this paper, we propose ***`Tem-Adapter`****, which enables the learning of temporal dynamics and complex semantics by a visual Temporal Aligner and a textual Semantic Aligner. Unlike conventional pretrained knowledge adaptation methods that only concentrate on the downstream task objective, the Temporal Aligner introduces an extra language-guided autoregressive task aimed at facilitating the learning of temporal dependencies, with the objective of predicting future states based on historical clues and language guidance that describes event progression. Besides, to reduce the semantic gap and adapt the textual representation for better event description, we introduce a Semantic Aligner that first designs a template to fuse question and answer pairs as event descriptions and then learns a Transformer decoder with the whole video sequence as guidance for refinement. We evaluate* ***`Tem-Adapter`*** *and different pre-train transferring methods on two VideoQA benchmarks, and the significant performance improvement demonstrates the effectiveness of our method.* [1]

1. Introduction

Video Question Answering (VideoQA) is a task that aims to answer natural language questions based on the information available in observed videos. It has attracted considerable attention recently due to its promise to develop interactive AI systems capable of communicating with the dynamic visual environment using natural language. Despite significant advancements in recent years, VideoQA remains a challenging problem that requires models to comprehensively understand and dynamically align the semantics of both the visual world and natural language.

[1]Our code can be found at: https://github.com/XLiu443/Tem-adapter
*Equal contribution.
†Corresponding author.

Recently, a variety of methods [79, 78, 82, 34, 74, 63] have demonstrated impressive results in utilizing the large-scale vision-language pre-trained (VLP) model to enhance downstream VideoQA tasks. These methods pre-train models by aligning the video and language domains and then apply them to VideoQA tasks via fine-tuning or even zero-shot learning. Nevertheless, pre-training large-scale models necessitate a large number of video-text pairs, *e.g.,* more than 10 million videos, and entails expensive computational costs. This motivates us to explore cheaper and lighter alternative pre-trained models.

Using image-VLP models is one potential option as they also align the semantics of vision and language domains. Compared to video-based models, image-based models offer two significant advantages. Firstly, training image-based models is less expensive when considering an equal number of data pairs. Secondly, image-VLP models have made significant progress in recent years, with some of them widely available and used, such as CLIP [56]. However, as depicted in Figure 1, domain gaps persist between image-based pre-training and VideoQA tasks since image-based models are unable to learn the temporal dynamics and corresponding complex event semantics of the visual world.

To address the challenge posed by the domain gaps, this paper proposes **`Tem-adapter`** , an adapter network that leverages the interaction between visual and textual modalities to learn temporal dynamics and complex semantics. Unlike existing methods that directly adapt visual and textual features with the objective of the downstream tasks, such as cross-modal matching in VideoQA, **`Tem-Adapter`** introduces an additional language-guided autoregressive task to facilitate learning of temporal dynamics, which enables visual representations to predict future states based on historical information and language guidance. Besides, we introduce cross-modal interactions to further reduce the semantic gap and refine textual representation.

Our approach consists of a visual Temporal Aligner and a textual Semantic Aligner. The Temporal Aligner uses a Transformer encoder to learn the temporal relations and refine visual features. To optimize it, we build an auto-regression model by a Transformer decoder to generate the future state with historical information. We also incorporate the language information for visual refinement by adding textual embeddings as the condition memory of the Transformer decoder. This auto-regression model is supervised by a reconstruction loss with ground truth future frames to encourage the learning of temporal dependencies. In addition to the visual branch, we introduce a Semantic Aligner for better event description by reducing the semantic gap between the original texts crawled from the net and question-answer pairs in VideoQA. First, we design templates to fuse textual questions and answers to generate declarative sentences to reduce the domain gap between training and downstream languages. Then, a Transformer decoder is employed to refine textual embeddings by incorporating the entire video sequence as a memory condition for video-text interactions.

We conduct experiments on two public VideoQA benchmarks including SUTD-TrafficQA [76] and MSR-VTT-MC [75], and evaluate different categories of methods that transfer the pre-trained knowledge to downstream tasks, such as other adapter models [12], prompt learning [89, 21], and fully or partial finetuning. The significant improvement demonstrates the effectiveness of **`Tem-adapter`** to adapt the image-language pre-trained model for VideoQA tasks.

2. Related Work

Video Question Answering. Compared with Image-based Question Answering, Video QA is more challenging due to the complex temporal dependency, which requires models to dynamically align the visual observation and natural language. Some previous methods [67, 3, 48, 81, 28, 65, 77, 27, 26, 4, 7, 6, 8, 41, 84, 91, 10, 11, 36, 13, 17, 22, 23, 51, 62, 72] learn the models with the given manually annotated datasets. Recently, more and more methods [79, 78, 82, 34, 74, 63, 29] propose to utilize the representation ability of large-scale pre-trained vision-language models to solve the VideoQA task. For example, VQA-T [78] pre-trains the vision language model with randomly sampled video-question-answer triplets in the HowTo100M [47] dataset. BiLM [79] pre-trains frozen bidirectional language models with some adapter layers on the WebVid10M [1] dataset and achieve excellent zero-shot VideoQA performance on the downstream tasks. However, pretraining the large-scale model is expensive, which motivates us to find a cheaper alternative manner. In this paper, we propose to use existing image-language pre-trained models, such as CLIP [56], to provide the alignment between the visual and textual domains. The most related method is ATP [2], which also applies CLIP as the pre-trained model to extract frame and language embeddings, and learns an atemporal probe to select the most discriminative frames to match the video and QA pairs. Different from ATP selecting frames to transfer the video to images, we propose **`Tem-adapter`** to learn the temporal dynamics and reduce the gap between the pre-trained image domain and downstream video domain.

Vision Language Pre-training. The goal of VLP is to help downstream tasks by pre-training the model on large-scale vision-text pairs [56, 37, 20]. It always crawls a huge amount of vision-text pairs from the web and trains the model in a self-supervised manner, such as reconstruction [40, 15, 9, 30], contrastive matching [56, 20, 18], or the combination of both two [38, 71, 25]. Different from image-VLP, video-VLP has more diverse proxy tasks, such as frame/language mask [66, 90], video-language matching [46], video/language generation [73, 45], and

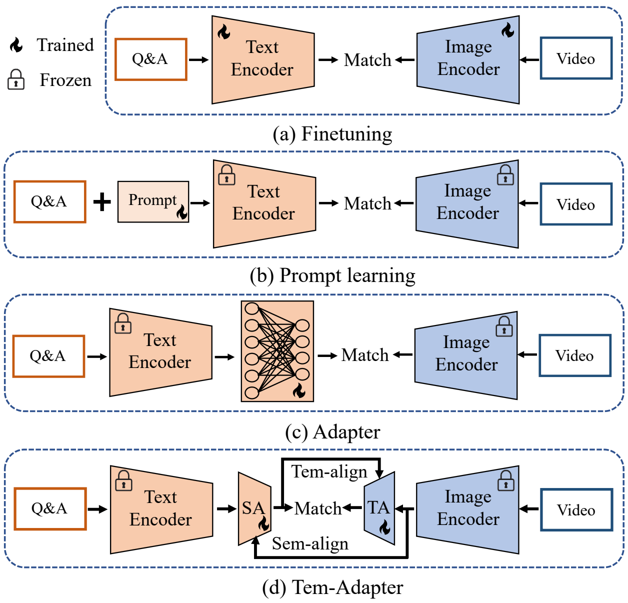

Figure 2: **Various methods for pre-trained knowledge adaptation in VideoQA .** (a) Fine-tuning updates pre-trained parameters with downstream training. (b) Prompt learning freezes the pre-trained model and introduces learnable textual tokens for adaptation. (c) Adapter-based methods add new adapter layers to transfer knowledge. (d) Our approach, `Tem-adapter` , leverages the visual-textual interactions to learn temporal dynamics to reduce the gap between image-based pre-training and video-based tasks.

frame/language order prediction [33, 39]. Recently, VLP models have shown great potential for other downstream applications, such as zero-shot and few-shot visual recognition [56, 12, 89, 88, 86], segmentation and detection [60, 87], image generation [50, 59, 52], and action understanding [70, 68]. In this paper, we use the CLIP model as our pre-trained model.

Pre-trained Knowledge Transfer. In light of the remarkable success of the large-scale pre-trained model, many methods have been proposed to solve the problem that how to efficiently adapt the pre-trained knowledge to the downstream tasks. Early methods finetune all parameters of the pre-trained model with the downstream objectives [14, 53]. However, with the model scale increasing, the cost of finetuning the full model is extremely expensive. Thus, an increasing number of methods focus on knowledge adaptation with light parameters. A straightforward way is to partially finetune the model parameters [49]. Besides, prompt learning methods are proposed recently to reformulate the downstream tasks as the pre-trained tasks and freeze all model parameters [43, 54, 58, 55]. The previous prompt-based methods [54, 55] design the template of prompt using the prior of natural human language. Then, some methods [24, 64] propose to automatically optimize the prompt template, and [42, 69, 44, 89, 5] further learn the prompts in the continuous embedding space where the discrete word constraint is relaxed. In addition, some methods freeze the model and train some extra adapter layers and also achieve the supersizing performance [12, 85, 1]. As shown in Figure 2, we illustrate the difference between `Tem-adapter` and other categories of methods transferring the image-based VLP model to the VideoQA task. Compared with other pre-train adaptation methods, our `Tem-adapter` proposes an autoregression task to reduce the temporal gap (tem-align) and introduce the cross-domain interactions (sem-align) to reduce the semantic gap between the pre-training and downstream tasks.

3. Approach

In this section, we first briefly introduce the VideoQA task and then present our approach, `Tem-adapter` , which introduces the cross-modal interactions to learn temporal dynamics and complex semantics. Finally, we give the details of our implementation.

3.1. Problem Definition

The goal of VideoQA is to find an answer $\hat{\mathbf{a}}$ from the answer pool, given the conditions of question $\mathbf{q}$ and observed video data $\mathbf{v}$ as below:

$$\hat{\mathbf{a}} = \arg\max_{\mathbf{a}\in\mathcal{A}} \mathcal{S}(\mathbf{a}|\mathbf{v},\mathbf{q}), \tag{1}$$

where S is a match function to generate the score of each answer candidate given the question and video content.

3.2. Tem-Adapter

Here, we propose `Tem-adapter` to learn temporal dynamics and complex semantics by the interactions of visual and textual domains to adapt pre-trained Image-VLP models to downstream VideoQA tasks. Figure 3 shows an overview of `Tem-Adapter`'s framework. In particular, `Tem-adapter` consists of two components, including a visual Temporal Aligner and a textual Semantic Aligner. Details for each component are as follows.

Temporal Aligner. The goal of the Temporal Aligner is to reduce the gap between image-based pre-training and video-based downstream tasks. To achieve this goal, we introduce an auxiliary language-guided autoregressive task to facilitate the learning of temporal dynamics.

Specifically, given a video $\mathbf{v}$, we apply the frozen CLIP [56] image encoder f to extract the feature of each frame as $f(\mathbf{v}) = \{f(\mathbf{v}_t)|_{t=1}^{T}\} \in \mathcal{R}^{T\times C}$, where T is the frame number of videos and C is the feature dimension of CLIP. Then we refine these features with our Temporal Aligner model which is built by a transformer encoder f_e with parameters ϕ to learn the temporal relations of frames.

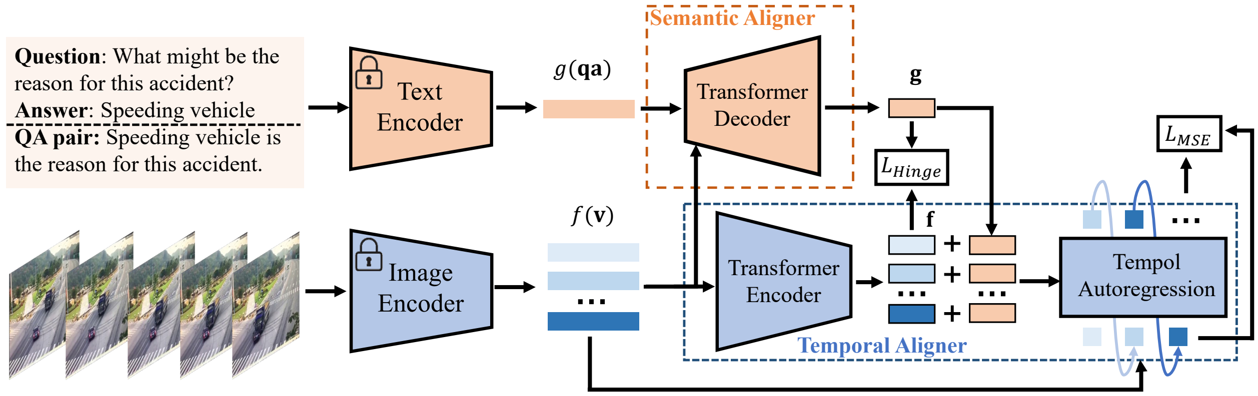

Figure 3: The overall framework of **`Tem-adapter`** . Our **`Tem-adapter`** consists of two modules, Semantic Aligner (orange) and Temporal Aligner (blue). Semantic Aligner takes the text embedding of template-based QA pair as the input, and applies a Transformer decoder to incorporate cross-modal interactions. The Temporal Aligner introduces a language-guided autoregressive task to learn temporal dynamics. The training is guided by classification and reconstruction losses.

The refined features can be represented as:

$$\mathbf{f} = f_e(f(\mathbf{v}); \phi), \tag{2}$$

which are with the same dimension $\mathcal{R}^{T\times C}$.

To guide the Temporal Aligner to learn temporal dynamics, we further build an autoregressive sequential model, following GPT [57], which generates the feature of the current frame with previous frames and guidance from textual information that describes event progression. Mathematically, it can be formulated as:

$$\hat{f}(\mathbf{v}_i) = f_d(f(\mathbf{v}_{1:i-1})|\mathbf{f}, \mathbf{g}; \psi), \tag{3}$$

where $f(\mathbf{v}_{1:i-1})$ is the input of the transformer decoder, which is achieved by applying an attention mask to focus on the history clues. $\mathbf{f}, \mathbf{g}$ are used as the guidance information, where $\mathbf{f}$ denotes latent visual embeddings obtained by Temporal Aligner and $\mathbf{g}$ (see equation 5) is the textual embeddings describing temporal event. $\mathbf{f}$ and $\mathbf{g}$ are first fused by adding and then serve as the memory key and value of the Transformer decoder. We apply a reconstruction loss to encourage the learning of temporal dependencies which is formulated as the distance between the results of the autoregression model $\hat{f}(\mathbf{v}_i)$ and the ground-truth $f(\mathbf{v}_i)$:

$$L_{MSE} = \sum_{i=1}^{n} ||f(\mathbf{v}_i) - \hat{f}(\mathbf{v}_i)||_2^2. \tag{4}$$

Semantic Aligner. In addition to the visual branch, we introduce a Semantic Aligner to adapt the textual representation for better event description and reduce the semantic gaps. We first design templates to align the question-answer pairs in VideoQA to declarative sentences, since the training data of CLIP is crawled from the web which always describes the image in a declarative manner. Then, we apply a Transformer decoder with the video sequence as the memory for video-text interactions.

Given a question $\mathbf{q}$ and a corresponding answer $\mathbf{a}$, we first apply the grammar parsing on the question $\mathbf{q}$ to obtain the subject-verb-object structure, and then we modify the order of the sentence to obtain a declarative sentence and leave the position of the interrogative pronouns as a mask '[]'. Finally, we fill in the default '[]' with the answer to update the QA pair as a refined event description. For instance, the question is *'Which might be the reason for this accident?'*, and the answer is *'Improper lane change'*. We first obtain a declarative sentence with the default as *'[ ] is the reason for this accident'*, and then we can fill in the answer to get *'Speeding vehicle is the reason for this accident.'* In addition, for the general question, we first change it to a declarative sentence and then convert the sentence to negative form if the answer is No.

With the refined sentence $\mathbf{qa}$ as the input, the frozen text encoder g of the CLIP model outputs the textual feature as $g(\mathbf{qa}) \in \mathcal{R}^{C}$. To further refine the textual feature using the video content, we apply a transformer decoder with the visual feature sequence as the memory key and value and add a residual link to avoid overfitting. Formally, it can be formulated as follows:

$$\mathbf{g} = g(\mathbf{qa}) + \lambda g_d(g(\mathbf{qa}), f(\mathbf{v}); \theta), \tag{5}$$

where $\mathbf{g}$ denotes the learned textual embedding, λ is a parameter of the residual link, and θ is the parameters of the transformer decoder g_d. Both λ and θ can be learned automatically in the training process. With the cross-attention module in the transformer decoder, we encourage the model to find key information from the video to refine the textual embedding for better event description.

Cross-modal Matching. Given the refined visual and textual embeddings $\mathbf{f}$ and $\mathbf{g}$, we instantiate the score function in equation 1 by cosine similarity as:

$$S(\mathbf{a}|\mathbf{v}, \mathbf{q}) = \mathrm{softmax}(\bar{\mathbf{f}}^T\mathbf{g}), \tag{6}$$

where $\bar{\mathbf{f}}$ is video representation using the average pool on all frames, $\mathbf{g}$ is the language embedding with given answer question pair ($\mathbf{a}$, $\mathbf{q}$). Then, we can obtain the answer to the question with the highest matching score.

Then we apply a standard multi-choice classification Hinge loss [19] integrated with the auto-regression loss for training. The overall loss function can be described as:

$$L = L_{Hinge}(S(\mathbf{a}|\mathbf{v}, \mathbf{q}), y) + \gamma L_{MSE}, \tag{7}$$

where γ is a ratio to balance two loss functions, and y is the ground truth label of the answers. Please note that our **`Tem-adapter`** consists of 3 learnable modules including the textual decoder θ, the visual encoder ϕ, and the visual decoder ψ. The Hinge loss is applied to optimize θ and ϕ, while the auto-regression loss L_{MSE} is used to optimize all three modules. In the inference, only textual decoder θ and visual encoder ϕ are used.

3.3. Implementation Details

In this subsection, we provide more implementation details of our approach for better reproduction.

Network Architecture. Figure 3 shows the framework of **`Tem-adapter`** . In our experiments, we use the frozen model CLIP (ViT-B32) as the text and image encoders. The Semantic Aligner is a one-layer Transformer decoder with 16 heads. There is a linear layer before the Transformer decoder which maps text embeddings and image embeddings to a lower dimension (128). Another linear projection layer is added after the Transformer decoder to map the reduced dimension back to 512. The Temporal Aligner is a light transformer encoder-decoder architecture. Both the encoder and decoder contain only one layer. We set the number of heads to 16 in the experiments.

Inference Details. In the inference (given 1 question, 1 video, and k answers), we first apply the template to fuse the question and k answers to obtain k QA pairs (e.g. $k = 4$). Then we feed the QA pairs and video into CLIP textual and visual encoders to obtain the initial features. After that, we apply the learned semantic and temporal aligners to refine these features. Please note that the temporal auto-regression module is not used in the inference. Finally, we obtain 1 video feature with k QA features and apply the cosine similarity as well as softmax to select the answer.

Hyper-parameters. We follow the same video frames sampling method with prior methods [76]. We uniformly select 8 frames over a long video sequence and then pick consecutive 16 frames which are centered on those 8 frames. Overall 128 frames are selected for each video. We also tried another sampling method: uniformly extracting 128 or 256 frames on a video sequence, and the final results do not support the sampling method very well. In the loss function, we set γ as 100 to encourage the model to learn the temporal dynamics. And the learnable parameter λ in the Semantic Aligner is set to be the same dimension (512) as the output embeddings of the CLIP model .

Training Details. We use the pre-trained CLIP model to extract text embeddings and image embeddings. The Adam [31] optimizer is utilized to optimize the training process. We set the initial learning rate to be $1e-4$ with a batch size of 128. And we let the learning rate decay with a factor of 0.5 for every 10 epochs. Our model is trained for 50 epochs on a single Tesla V100 GPU and the training takes 8 hours to finish. Our model is implemented in PyTorch 1.8.0.

4. Experiments

In this section, we first benchmark our **`Tem-adapter`** and different categories of methods that transfer the pre-trained model into downstream tasks. Then we provide ablation studies to investigate the effect of each component.

4.1. Datasets & Experimental Protocols

SUTD-TrafficQA. SUTD-TrafficQA focuses on the question-answering task in the traffic situation, which requires the model to understand the traffic event and underlying causal relation. It consists of more than 10k videos with different traffic events and provides over 62,535 human-annotated QA pairs. Among them, 56,460 QA pairs are used for training and the rest 6,075 QA pairs are used for testing. This dataset highlights the cognitive capability of video event understanding models in complex traffic scenarios. It provides 6 challenging traffic-related reasoning tasks including "Basic understanding", "Event forecasting", "Reverse reasoning", "Counterfactual inference", "Introspection", "Attribution". All tasks are formulated as multiple-choice without limiting the number of candidate answers.

MSR-VTT-MC. MSR-VTT is collected and released mainly for the text-to-video retrieval task. Following [34] and [2], we use the multi-choice test setting to benchmark the VideoQA methods. The MSR-VTT-MC test set consists of 2990 QA pairs. Each video corresponds to five answers, with the original caption as the correct answer. We use the standard protocol with prior works in our experiments, where 7010 train+val video sequences and 140,200 QA pairs are utilized for training. Each video only contains one caption. We assign this caption as the correct answer for our question-answering task. To formulate four negative answers during training, we randomly select four captions from other captions that do not belong to the current video.

4.2. Baseline Methods

To investigate different categories of methods that transfer the pre-trained model into downstream tasks, such as finetuning, prompt learning, and adapter. We compare **`Tem-adapter`** with the other 10 baseline methods including Unsupervised CLIP [56], Unsupervised CLIP [56]

Table 1: VideoQA accuracies of baseline methods on both SUTD-TrafficQA and MSR-VTT-MC . **B**: "Basic understanding", **F**: "Forecasting task", **R**: "Reverse Reasoning", **C**: "Counterfactual inference", **I**: "Introspection", **A**: "Attribution". (C) and (A) denote strategies for adding prompts. (C) and (C*) denote training prompts with/without adapter heads.

Methods	**SUTD-TrafficQA**							**MSR-VTT-MC**
	B	**F**	**R**	**C**	**I**	**A**	**Avg**	**Acc**
Unsupervised CLIP [56]	25.6	20.1	34.0	30.8	22.8	28.8	26.5	74.1
CLIP [56] + Template	31.8	36.0	29.9	71.8	22.1	33.4	32.3	74.1
Totally finetuning	39.8	35.1	46.6	45.6	37.2	40.5	40.3	89.0
Partially finetuning	41.6	37.8	44.6	50.0	33.1	41.7	41.7	88.5
LORA [16]	38.7	38.7	36.7	37.9	34.5	38.1	38.3	85.4
CLIP-Adapter [12]	35.8	32.0	35.4	42.3	33.1	32.1	34.8	83.4
Multi-layer Adapter [12]	30.5	26.6	26.5	38.5	28.3	25.8	29.1	90.7
Prompt learning (C) [89]	42.4	32.4	45.2	55.5	40.7	43.6	42.9	89.0
Prompt learning (C*) [89]	40.3	33.2	41.0	46.5	34.9	38.4	39.7	90.8
Prompt learning (A) [21]	41.7	31.5	40.1	48.4	33.1	41.4	41.1	88.0
Tem-adapter	**46.0**	**36.5**	**44.6**	**55.0**	**34.5**	**47.7**	**46.0**	**94.3**

Table 2: Ablation studies on the SUTD-TrafficQA and MSR-VTT-MC datasets. Tp, SA, TA, and Ar denote Template, Semantic Aligner, Temporal Aligner, and Auto-regression, respectively. No ablation studies ("\") concentrate on Tp, as it is not compatible with the MSR-VTT-MC dataset.

Methods	**SUTD-TrafficQA**	**MSR-VTT-MC**
	Avg	**Acc**
CLIP-Adapter [12]	34.8	83.4
w/o Tp	43.9	\
w/o SA	44.2	91.8
w/o TA	43.6	91.6
w/o Ar	44.9	92.6
w/o TA and SA	32.3	74.1
Tem-adapter	**46.0**	**94.3**

+ Language template, Totally fine-tuning, Partially fine-tuning, LORA [16], CLIP-Adapter [12], Multi-layer CLIP-Adapter [12], Prompt learning (change words) [89] (with/without using adapter heads), and Prompt learning (add words) [21]. The detailed descriptions are shown in the **supplementary materials**.

4.3. Benchmarking against Baseline Methods

In this subsection, we compare and discuss our approach's performance against the above-mentioned baselines on both SUTD-TrafficQA and MSR-VTT-MC datasets. The results of all methods on both datasets are summarized in Table 1. We can draw conclusions as follows from the results.

CLIP can obtain good zero-shot results. With our language template, the clip model can achieve 32.3% accuracy on the SUTD-TrafficQA dataset, which is almost close to the previous state-of-the-art performance (37.1%). It shows the strong representation ability of the CLIP model.

More parameters $\neq$ Better performance. We have two groups of experiments that focus on the number of parameters, including totally/partially finetuning and single-layer/multi-layer Adapters. We obtain contradictory conclusions from different datasets. On SUTD-TrafficQA, the model with fewer parameters works better, while on MSR-VTT-MC, the model with more parameters has better performance. It is not surprising since the scale of training data is different. MSR-VTT-MC has more training data and can support training more parameters. We also compare the learnable parameters of adapter-based methods in Table 5. **Tem-adapter** achieve better performance with fewer parameters than Multi-layer Adapter. Compared with the prompt-learning-based method, our method needs more parameters, but its inference is not time-consuming.

Different categories of methods are comparable. Balancing the performance on both datasets, different categories of methods that transfer the image-based VLP model to VideoQA don't have clear performance differences. Adapter-based methods obtain poor results on SUTD-TrafficQA, but multi-layer Adapter works well on MSR-VTT-MC. Besides, by comparing the prompt learning methods with/without adapter heads, we found that the prompt and adapter can be joint learning to obtain better performance.

Cross-modal interaction and temporal dynamics matter. Compared with CLIP-Adapter (both single-layer and multi-layer ones), our **Tem-adapter** achieves more than 10% improvement on SUTD-TrafficQA. The key difference between **Tem-adapter** and other adapter-based methods is that we introduce temporal alignment to reduce the temporal gap and apply the cross-modal interaction to refine the representation and reduce the semantic gap. **Tem-adapter** outperforms other methods by a large mar-

Table 3: Comparison with the state-of-the-art methods on the SUTD-TrafficQA dataset.

Methods	Accuracy	Year
Human	95.4	–
VIS+LSTM [61]	29.9	2015
TVQA [35]	35.2	2018
HCRN [32]	36.5	2020
BERT-VQA [80]	33.7	2020
Eclipse [76]	37.1	2021
CLIP [56]	32.3	2021
CLIP+HCRN[56, 32]	41.9	2021
ATP [2]	35.6	2022
`Tem-adapter`	**46.0**	

gin on both two datasets. For example, it obtains 3.1% improvement than Prompt learning (C) on SUTD-TrafficQA and 3.6% than Multi-layer Adapter on MSR-VTT-MC.

4.4. Comparison against State-of-the-Art Methods

SUTD-TrafficQA. On SUTD-TrafficQA, we compare our proposed method with state-of-the-art methods, including [32, 76, 61, 35, 80]. The results are summarized in Table 3. We can observe that our **`Tem-adapter`** outperforms other methods by a large margin. For example, compared with HCRN [32], **`Tem-adapter`** achieves almost 10% improvement. By using the CLIP model to extract visual features, HCRN+CLIP can obtain 41.9% accuracy, which is better than using the original HCRN. Besides, we show that the performance improvement is not only from the better representation ability of the CLIP model since our **`Tem-adapter`** further improve HCRN+CLIP with +4.1% accuracy. In addition, we also significantly outperform the ATP method, which also uses the CLIP model. This significant performance improvement demonstrates the effectiveness of **`Tem-adapter`** .

MSR-VTT-MC. On MSR-VTT-MC, we compare **`Tem-adapter`** with some recent state-of-the-art methods, including ActBERT[90], ClipBERT[34], MERLOT [83], VideoCLIP [66], CLIP [56], and ATP [2]. Results are shown in Table 4. First, we can find that the CLIP model benefits the performance on the MSR-VTT-MC dataset. All methods using the CLIP model, such as CLIP [56], ATP [2], and **`Tem-adapter`** achieve good performance. ATP [2] learns an atemporal probe to select a frame from a video sequence to obtain the visual embedding and achieve good performance on MSR-VTT-MC. Compared with it, **`Tem-adapter`** can further obtain 94.3% accuracy and 1.1% relative improvement.

4.5. Ablation Studies

We provide ablation studies to investigate the effect of each component. Four main modules in our **`Tem-adapter`** are evaluated, including Temporal Aligner, Semantic Aligner, language template, and auto-regression.

Table 4: Comparison with the state-of-the-art methods on the MSR-VTT-MC dataset.

Methods	Accuracy	Year
VideoCLIP [66]	92.1	2019
ActBERT[90]	85.7	2020
ClipBERT[34]	88.2	2021
MERLOT [83]	90.9	2021
CLIP [56]	74.1	2021
ATP [2]	93.2	2022
`Tem-adapter`	**94.3**	

As shown in Table 2, we provide the performance of different variants of **`Tem-adapter`** and a baseline adapter method on each task in the SUTD-TrafficQA and MSR-VTT-MC datasets. The baseline method is CLIP-adapter, which also uses the language template and learns a linear adapter layer without cross-domain interaction. We observe that all variants of **`Tem-adapter`** can achieve significant improvement over the baseline method, which shows that the interaction across visual and textual is very important for video analysis. Besides, compared with **`Tem-adapter`** , we can observe a clear performance drop when we remove each component, which demonstrates all of them are indispensable. When both temporal and semantic aligners are removed, the performance drops dramatically.

In addition, we also conducted an experiment to show whether the Temporal Aligner learns the temporal dynamic information. Table 6 shows the reconstruction quality of our methods with/without language guidance ($\boldsymbol{g}$) and with/without Temporal Aligner ($\boldsymbol{f}$). We can observe that both the language guidance and Temporal Aligner assist in the learning of temporal dynamics.

Furthermore, we also provide some parameter analysis experiments about our network design. And evaluate **`Tem-adapter`** on different image-text pre-train models. Please kindly refer to the **supplementary materials** for the detailed results and analysis.

4.6. Visualization

In this subsection, we provide some qualitative results to analyze our approach. As shown in Figure 4, we show some visualization examples of VdieoQA on the SUTD-TrafficQA dataset. In the top raw, we plot two positive examples. We can observe that our **`Tem-adapter`** learns the temporal dynamics with the cross-domain interactions to obtain a better understanding of video events. Besides, we also provide some failure cases to help explore the boundary of our approach. We find that **`Tem-adapter`** can not work well for situations that need complex reasoning, especially

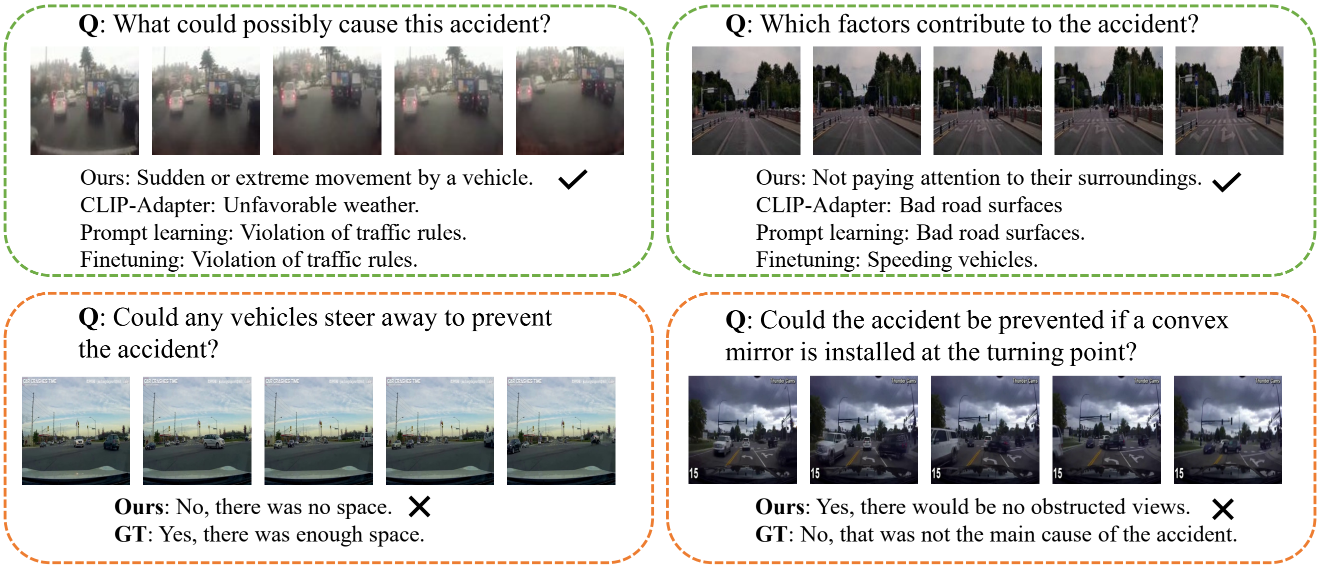

Figure 4: Visualization of some examples of the VideoQA task from the SUTD-TrafficQA dataset. The top row shows some positive examples, where **`Tem-adapter`** learns temporal dynamics to understand the traffic event and give the correct answers. The bottom row shows some failed cases. Despite better matching between visual and textual domains, our approach fails on complex problems which require reasoning ability.

Table 5: The comparisons of our method and other baseline methods on model parameters (M), inference time (ms/video), and accuracy in the SUTD-TrafficQA dataset.

Metrics	LORA [16]	Prompt learning (C) [89]	Prompt learning (C*) [89]	CLIP-Adapter [12]	Multi-layer Adapter [12]	Ours
Parameters	1.44	3.55	0.77	1.05	4.85	3.62
Inference time	5.37	5.33	5.17	5.13	5.14	5.32
Accuracy	38.3	42.9	39.7	34.8	29.1	**46.0**

Table 6: The comparison of reconstruction ability. We use peak signal-to-noise ratio (PSNR) as the metric.

	`Tem-adapter`	w/o g	w/o **f** and g
PSNR	29.1	28.7	27.4

for the underlying cause-effect relations. We leave it as a future work to enhance the reasoning ability of our model. Besides, we also provide more examples of our language templates in the **supplementary materials**.

5. Conclusion

In this paper, we proposed **`Tem-adapter`** to adapt the pre-trained image-language model to the downstream VideoQA task. We jointly learn the Temporal Aligner and Semantic Aligner through cross-modal interactions. **`Tem-adapter`** introduces a language-guided autoregressive task to guide the learning of temporal dependency and thus reduce the temporal gap between image-based pre-train and video-based QA tasks. Besides, we utilize a rule-based template and video-based interaction to refine textual representation and reduce semantic gaps. We validate the effectiveness of our approach on four VideoQA benchmarks.

Broader Impacts and Limitations. Understanding how the big model adaptation works on VideoQA is of great help in developing general foundation models and interactive AI. However, our work also has certain limitations. For example, since extracting video features online takes up too many computing resources, we extract their features offline and fix them. We leave it as future work to learn the adapter within the backbone network.

Acknowledgment

This work was supported in part by the National Natural Science Foundation of China under Grant 62206153, Young Elite Scientists Sponsorship Program by CAST (No. 2023QNRC002), by the Tsinghua Shenzhen International Graduate School-Shenzhen Pengrui Endowed Professorship Scheme of Shenzhen Pengrui Foundation, by the UKRI grant: Turing AI Fellowship EP/W002981/1, by the National Institutes of Health (NIH) under Contract R01HL159805, by the NSF-Convergence Accelerator Track-D award #2134901, by a grant from Apple Inc., a grant from KDDI Research Inc, and generous gifts from Salesforce Inc., Microsoft Research, and Amazon Research. We would also like to thank the Royal Academy of Engineering and FiveAI.

References

[1] Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman. Frozen in time: A joint video and image encoder for end-to-end retrieval. In *ICCV*, pages 1728–1738, 2021. 2, 3

[2] Shyamal Buch, Cristóbal Eyzaguirre, Adrien Gaidon, Jiajun Wu, Li Fei-Fei, and Juan Carlos Niebles. Revisiting the" video" in video-language understanding. In *CVPR*, pages 2917–2927, 2022. 2, 5, 7

[3] Santiago Castro, Mahmoud Azab, Jonathan Stroud, Cristina Noujaim, Ruoyao Wang, Jia Deng, and Rada Mihalcea. Lifeqa: A real-life dataset for video question answering. In *LREC*, pages 4352–4358, 2020. 2

[4] Aman Chadha, Gurneet Arora, and Navpreet Kaloty. iperceive: Applying common-sense reasoning to multi-modal dense video captioning and video question answering. *arXiv preprint arXiv:2011.07735*, 2020. 2

[5] Guangyi Chen, Weiran Yao, Xiangchen Song, Xinyue Li, Yongming Rao, and Kun Zhang. Plot: Prompt learning with optimal transport for vision-language models. In *ICLR*, 2023. 3

[6] Seongho Choi, Kyoung-Woon On, Yu-Jung Heo, Ahjeong Seo, Youwon Jang, Minsu Lee, and Byoung-Tak Zhang. Dramaqa: Character-centered video story understanding with hierarchical qa. In *AAAI*, volume 35, pages 1166–1174, 2021. 2

[7] Anthony Colas, Seokhwan Kim, Franck Dernoncourt, Siddhesh Gupte, Daisy Zhe Wang, and Doo Soon Kim. Tutorialvqa: Question answering dataset for tutorial videos. *arXiv preprint arXiv:1912.01046*, 2019. 2

[8] Long Hoang Dang, Thao Minh Le, Vuong Le, and Truyen Tran. Object-centric representation learning for video question answering. In *IJCNN*, pages 1–8, 2021. 2

[9] Zi-Yi Dou, Yichong Xu, Zhe Gan, Jianfeng Wang, Shuohang Wang, Lijuan Wang, Chenguang Zhu, Zicheng Liu, Michael Zeng, et al. An empirical study of training end-to-end vision-and-language transformers. *arXiv preprint arXiv:2111.02387*, 2021. 2

[10] Chenyou Fan, Xiaofan Zhang, Shu Zhang, Wensheng Wang, Chi Zhang, and Heng Huang. Heterogeneous memory enhanced multimodal attention model for video question answering. In *CVPR*, pages 1999–2007, 2019. 2

[11] Jiyang Gao, Runzhou Ge, Kan Chen, and Ram Nevatia. Motion-appearance co-memory networks for video question answering. In *CVPR*, pages 6576–6585, 2018. 2

[12] Peng Gao, Shijie Geng, Renrui Zhang, Teli Ma, Rongyao Fang, Yongfeng Zhang, Hongsheng Li, and Yu Qiao. Clip-adapter: Better vision-language models with feature adapters. *arXiv preprint arXiv:2110.04544*, 2021. 2, 3, 6, 8

[13] Noa Garcia, Mayu Otani, Chenhui Chu, and Yuta Nakashima. Knowit vqa: Answering knowledge-based questions about videos. In *AAAI*, volume 34, pages 10826–10834, 2020. 2

[14] Ross Girshick. Fast r-cnn. In *ICCV*, pages 1440–1448, 2015. 3

[15] Yicong Hong, Qi Wu, Yuankai Qi, Cristian Rodriguez-Opazo, and Stephen Gould. Vln bert: A recurrent vision-and-language bert for navigation. In *CVPR*, pages 1643–1653, 2021. 2

[16] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021. 6, 8

[17] Deng Huang, Peihao Chen, Runhao Zeng, Qing Du, Mingkui Tan, and Chuang Gan. Location-aware graph convolutional networks for video question answering. In *AAAI*, volume 34, pages 11021–11028, 2020. 2

[18] Aashi Jain, Mandy Guo, Krishna Srinivasan, Ting Chen, Sneha Kudugunta, Chao Jia, Yinfei Yang, and Jason Baldridge. Mural: multimodal, multitask retrieval across languages. *arXiv preprint arXiv:2109.05125*, 2021. 2

[19] Yunseok Jang, Yale Song, Youngjae Yu, Youngjin Kim, and Gunhee Kim. Tgif-qa: Toward spatio-temporal reasoning in visual question answering. In *CVPR*, pages 2758–2766, 2017. 5

[20] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *ICML*, pages 4904–4916, 2021. 2

[21] Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. Visual prompt tuning. *arXiv preprint arXiv:2203.12119*, 2022. 2, 6

[22] Jianwen Jiang, Ziqiang Chen, Haojie Lin, Xibin Zhao, and Yue Gao. Divide and conquer: Question-guided spatio-temporal contextual attention for video question answering. In *AAAI*, volume 34, pages 11101–11108, 2020. 2

[23] Pin Jiang and Yahong Han. Reasoning with heterogeneous graph alignment for video question answering. In *AAAI*, volume 34, pages 11109–11116, 2020. 2

[24] Zhengbao Jiang, Frank F Xu, Jun Araki, and Graham Neubig. How can we know what language models know? *TACL*, 8:423–438, 2020. 3

[25] Aishwarya Kamath, Mannat Singh, Yann LeCun, Gabriel Synnaeve, Ishan Misra, and Nicolas Carion. Mdetr-modulated detection for end-to-end multi-modal understanding. In *ICCV*, pages 1780–1790, 2021. 2

[26] Hyounghun Kim, Zineng Tang, and Mohit Bansal. Dense-caption matching and frame-selection gating for temporal localization in videoqa. *arXiv preprint arXiv:2005.06409*, 2020. 2

[27] Junyeong Kim, Minuk Ma, Trung Pham, Kyungsu Kim, and Chang D Yoo. Modality shifting attention network for multi-modal video question answering. In *CVPR*, pages 10106–10115, 2020. 2

[28] Kyung-Min Kim, Min-Oh Heo, Seong-Ho Choi, and Byoung-Tak Zhang. Deepstory: Video story qa by deep embedded memory networks. *arXiv preprint arXiv:1707.00836*, 2017. 2

[29] Seonhoon Kim, Seohyeong Jeong, Eunbyul Kim, Inho Kang, and Nojun Kwak. Self-supervised pre-training and contrastive representation learning for multiple-choice video qa. In *AAAI*, volume 35, pages 13171–13179, 2021. 2

[30] Wonjae Kim, Bokyung Son, and Ildoo Kim. Vilt: Vision-and-language transformer without convolution or region supervision. In *ICML*, pages 5583–5594, 2021. 2

[31] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 5

[32] Thao Minh Le, Vuong Le, Svetha Venkatesh, and Truyen Tran. Hierarchical conditional relation networks for video question answering. In *CVPR*, pages 9972–9981, 2020. 7

[33] Chenyi Lei, Shixian Luo, Yong Liu, Wanggui He, Jiamang Wang, Guoxin Wang, Haihong Tang, Chunyan Miao, and Houqiang Li. Understanding chinese video and language via contrastive multimodal pre-training. In *ACM MM*, pages 2567–2576, 2021. 3

[34] Jie Lei, Linjie Li, Luowei Zhou, Zhe Gan, Tamara L Berg, Mohit Bansal, and Jingjing Liu. Less is more: Clipbert for video-and-language learning via sparse sampling. In *CVPR*, pages 7331–7341, 2021. 2, 5, 7

[35] Jie Lei, Licheng Yu, Mohit Bansal, and Tamara L Berg. Tvqa: Localized, compositional video question answering. *arXiv preprint arXiv:1809.01696*, 2018. 7

[36] Jie Lei, Licheng Yu, Tamara L Berg, and Mohit Bansal. Tvqa+: Spatio-temporal grounding for video question answering. *arXiv preprint arXiv:1904.11574*, 2019. 2

[37] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. *arXiv preprint arXiv:2201.12086*, 2022. 2

[38] Junnan Li, Ramprasaath Selvaraju, Akhilesh Gotmare, Shafiq Joty, Caiming Xiong, and Steven Chu Hong Hoi. Align before fuse: Vision and language representation learning with momentum distillation. *NeurIPS*, 34, 2021. 2

[39] Linjie Li, Yen-Chun Chen, Yu Cheng, Zhe Gan, Licheng Yu, and Jingjing Liu. Hero: Hierarchical encoder for video+ language omni-representation pre-training. *arXiv preprint arXiv:2005.00200*, 2020. 3

[40] Liunian Harold Li, Mark Yatskar, Da Yin, Cho-Jui Hsieh, and Kai-Wei Chang. Visualbert: A simple and performant baseline for vision and language. *arXiv preprint arXiv:1908.03557*, 2019. 2

[41] Xiangpeng Li, Jingkuan Song, Lianli Gao, Xianglong Liu, Wenbing Huang, Xiangnan He, and Chuang Gan. Beyond rnns: Positional self-attention with co-attention for video question answering. In *AAAI*, volume 33, pages 8658–8665, 2019. 2

[42] Xiang Lisa Li and Percy Liang. Prefix-tuning: Optimizing continuous prompts for generation. *arXiv preprint arXiv:2101.00190*, 2021. 3

[43] Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *arXiv preprint arXiv:2107.13586*, 2021. 3

[44] Xiao Liu, Yanan Zheng, Zhengxiao Du, Ming Ding, Yujie Qian, Zhilin Yang, and Jie Tang. Gpt understands, too. *arXiv preprint arXiv:2103.10385*, 2021. 3

[45] Huaishao Luo, Lei Ji, Botian Shi, Haoyang Huang, Nan Duan, Tianrui Li, Jason Li, Taroon Bharti, and Ming Zhou. Univl: A unified video and language pre-training model for multimodal understanding and generation. *arXiv preprint arXiv:2002.06353*, 2020. 2

[46] Antoine Miech, Jean-Baptiste Alayrac, Lucas Smaira, Ivan Laptev, Josef Sivic, and Andrew Zisserman. End-to-end learning of visual representations from uncurated instructional videos. In *CVPR*, pages 9879–9889, 2020. 2

[47] Antoine Miech, Dimitri Zhukov, Jean-Baptiste Alayrac, Makarand Tapaswi, Ivan Laptev, and Josef Sivic. Howto100m: Learning a text-video embedding by watching hundred million narrated video clips. In *ICCV*, pages 2630–2640, 2019. 2

[48] Jonghwan Mun, Paul Hongsuck Seo, Ilchae Jung, and Bohyung Han. Marioqa: Answering questions by watching gameplay videos. In *ICCV*, pages 2867–2875, 2017. 2

[49] Akihiro Nakamura and Tatsuya Harada. Revisiting fine-tuning for few-shot learning. *arXiv preprint arXiv:1910.00216*, 2019. 3

[50] Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. *arXiv preprint arXiv:2112.10741*, 2021. 3

[51] Jungin Park, Jiyoung Lee, and Kwanghoon Sohn. Bridge to answer: Structure-aware graph interaction network for video question answering. In *CVPR*, pages 15526–15535, 2021. 2

[52] Or Patashnik, Zongze Wu, Eli Shechtman, Daniel Cohen-Or, and Dani Lischinski. Styleclip: Text-driven manipulation of stylegan imagery. In *ICCV*, pages 2085–2094, 2021. 3

[53] Matthew E Peters, Sebastian Ruder, and Noah A Smith. To tune or not to tune? adapting pretrained representations to diverse tasks. *arXiv preprint arXiv:1903.05987*, 2019. 3

[54] Fabio Petroni, Tim Rocktäschel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, Alexander H Miller, and Sebastian Riedel. Language models as knowledge bases? *arXiv preprint arXiv:1909.01066*, 2019. 3

[55] Nina Poerner, Ulli Waltinger, and Hinrich Schütze. Bert is not a knowledge base (yet): Factual knowledge vs. name-based reasoning in unsupervised qa. *arXiv preprint arXiv:1911.03681*, 2019. 3

[56] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, pages 8748–8763. PMLR, 2021. 2, 3, 5, 6, 7

[57] Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. Improving language understanding by generative pre-training. 2018. 4

[58] 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. 3

[59] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022. 3

[60] Yongming Rao, Wenliang Zhao, Guangyi Chen, Yansong Tang, Zheng Zhu, Guan Huang, Jie Zhou, and Jiwen Lu. Denseclip: Language-guided dense prediction with context-aware prompting. *arXiv preprint arXiv:2112.01518*, 2021. 3

[61] Mengye Ren, Ryan Kiros, and Richard Zemel. Exploring models and data for image question answering. *Advances in neural information processing systems*, 28, 2015. 7

[62] Arka Sadhu, Kan Chen, and Ram Nevatia. Video question answering with phrases via semantic roles. *arXiv preprint arXiv:2104.03762*, 2021. 2

[63] Paul Hongsuck Seo, Arsha Nagrani, and Cordelia Schmid. Look before you speak: Visually contextualized utterances. In *CVPR*, pages 16877–16887, 2021. 2

[64] Taylor Shin, Yasaman Razeghi, Robert L Logan IV, Eric Wallace, and Sameer Singh. Autoprompt: Eliciting knowledge from language models with automatically generated prompts. *arXiv preprint arXiv:2010.15980*, 2020. 3

[65] Xiaomeng Song, Yucheng Shi, Xin Chen, and Yahong Han. Explore multi-step reasoning in video question answering. In *ACM MM*, pages 239–247, 2018. 2

[66] Chen Sun, Austin Myers, Carl Vondrick, Kevin Murphy, and Cordelia Schmid. Videobert: A joint model for video and language representation learning. In *ICCV*, pages 7464–7473, 2019. 2, 7

[67] Makarand Tapaswi, Yukun Zhu, Rainer Stiefelhagen, Antonio Torralba, Raquel Urtasun, and Sanja Fidler. Movieqa: Understanding stories in movies through question-answering. In *CVPR*, pages 4631–4640, 2016. 2

[68] Guy Tevet, Brian Gordon, Amir Hertz, Amit H Bermano, and Daniel Cohen-Or. Motionclip: Exposing human motion generation to clip space. *arXiv preprint arXiv:2203.08063*, 2022. 3

[69] Maria Tsimpoukelli, Jacob L Menick, Serkan Cabi, SM Eslami, Oriol Vinyals, and Felix Hill. Multimodal few-shot learning with frozen language models. *NeurIPS*, 34:200–212, 2021. 3

[70] Mengmeng Wang, Jiazheng Xing, and Yong Liu. Actionclip: A new paradigm for video action recognition. *arXiv preprint arXiv:2109.08472*, 2021. 3

[71] Wenhui Wang, Hangbo Bao, Li Dong, and Furu Wei. Vlmo: Unified vision-language pre-training with mixture-of-modality-experts. *arXiv preprint arXiv:2111.02358*, 2021. 2

[72] Junbin Xiao, Xindi Shang, Angela Yao, and Tat-Seng Chua. Next-qa: Next phase of question-answering to explaining temporal actions. In *CVPR*, pages 9777–9786, 2021. 2

[73] Hu Xu, Gargi Ghosh, Po-Yao Huang, Prahal Arora, Masoumeh Aminzadeh, Christoph Feichtenhofer, Florian Metze, and Luke Zettlemoyer. Vlm: Task-agnostic video-language model pre-training for video understanding. *arXiv preprint arXiv:2105.09996*, 2021. 2

[74] Hu Xu, Gargi Ghosh, Po-Yao Huang, Dmytro Okhonko, Armen Aghajanyan, Florian Metze, Luke Zettlemoyer, and Christoph Feichtenhofer. Videoclip: Contrastive pre-training for zero-shot video-text understanding. *EMNLP*, 2021. 2

[75] Jun Xu, Tao Mei, Ting Yao, and Yong Rui. Msr-vtt: A large video description dataset for bridging video and language. In *CVPR*, pages 5288–5296, 2016. 2

[76] Li Xu, He Huang, and Jun Liu. Sutd-trafficqa: A question answering benchmark and an efficient network for video reasoning over traffic events. In *CVPR*, pages 9878–9888, 2021. 2, 5, 7

[77] Hongyang Xue, Wenqing Chu, Zhou Zhao, and Deng Cai. A better way to attend: Attention with trees for video question answering. *IEEE TIP*, 27(11):5563–5574, 2018. 2

[78] Antoine Yang, Antoine Miech, Josef Sivic, Ivan Laptev, and Cordelia Schmid. Just ask: Learning to answer questions from millions of narrated videos. In *ICCV*, pages 1686–1697, 2021. 2

[79] Antoine Yang, Antoine Miech, Josef Sivic, Ivan Laptev, and Cordelia Schmid. Zero-shot video question answering via frozen bidirectional language models. *NeurIPS*, 2022. 2

[80] Zekun Yang, Noa Garcia, Chenhui Chu, Mayu Otani, Yuta Nakashima, and Haruo Takemura. Bert representations for video question answering. In *WACV*, pages 1556–1565, 2020. 7

[81] Yunan Ye, Zhou Zhao, Yimeng Li, Long Chen, Jun Xiao, and Yueting Zhuang. Video question answering via attribute-augmented attention network learning. In *ACM SIGIR*, pages 829–832, 2017. 2

[82] Rowan Zellers, Jiasen Lu, Ximing Lu, Youngjae Yu, Yanpeng Zhao, Mohammadreza Salehi, Aditya Kusupati, Jack Hessel, Ali Farhadi, and Yejin Choi. Merlot reserve: Neural script knowledge through vision and language and sound. In *CVPR*, pages 16375–16387, 2022. 2

[83] Rowan Zellers, Ximing Lu, Jack Hessel, Youngjae Yu, Jae Sung Park, Jize Cao, Ali Farhadi, and Yejin Choi. Merlot: Multimodal neural script knowledge models. *NeurIPS*, 34:23634–23651, 2021. 7

[84] Zheng-Jun Zha, Jiawei Liu, Tianhao Yang, and Yongdong Zhang. Spatiotemporal-textual co-attention network for video question answering. *ACM TOMM*, 15(2s):1–18, 2019. 2

[85] Renrui Zhang, Rongyao Fang, Peng Gao, Wei Zhang, Kunchang Li, Jifeng Dai, Yu Qiao, and Hongsheng Li. Tip-adapter: Training-free clip-adapter for better vision-language modeling. In *CVPR*, 2022. 3

[86] Renrui Zhang, Longtian Qiu, Wei Zhang, and Ziyao Zeng. Vt-clip: Enhancing vision-language models with visual-guided texts. *arXiv preprint arXiv:2112.02399*, 2021. 3

[87] Chong Zhou, Chen Change Loy, and Bo Dai. Denseclip: Extract free dense labels from clip. *arXiv preprint arXiv:2112.01071*, 2021. 3

[88] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Conditional prompt learning for vision-language models. In *CVPR*, 2022. 3

[89] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *IJCV*, 2022. 2, 3, 6, 8

[90] Linchao Zhu and Yi Yang. Actbert: Learning global-local video-text representations. In *CVPR*, pages 8746–8755, 2020. 2, 7

[91] Yueting Zhuang, Dejing Xu, Xin Yan, Wenzhuo Cheng, Zhou Zhao, Shiliang Pu, and Jun Xiao. Multichannel attention refinement for video question answering. *ACM TOMM*, 16(1s):1–23, 2020. 2