The 28th Meeting on Image Recognition and Understanding

Visual Adapter for Extracting Textually-related Features for Video Captioning

Junan Chen[1,a)] Trung Thanh Nguyen[1,b)] Takahiro Komamizu[2,1,c)]
Ichiro Ide[1,2,d)]

Abstract

Recent advancements in video captioning have been driven by large-scale pretrained models, which follow the standard "pre-training followed by fine-tuning" paradigm, where the full model is fine-tuned for the downstream tasks. While effective, this approach becomes computationally prohibitive as the model size increases. The Parameter-Efficient Fine-Tuning (PEFT) approach offers a promising alternative but primarily focuses on the language components of Multi-modal Large Language Models (MLLMs). Despite recent progress, PEFT remains underexplored in multimodal tasks such as video captioning and lacks sufficient understanding of visual information during model fine-tuning. To bridge this gap, we introduce **Q**uery-**Adapter** (**Q-Adapter**), a lightweight visual adapter module designed to enhance MLLMs through efficient fine-tuning for the video captioning task. Q-Adapter integrates learnable query tokens and a gating layer into the Vision Encoder, enabling effective extraction of sparse, caption-relevant features without relying on external textual supervision. We evaluate Q-Adapter on the MSR-VTT benchmark, achieving state-of-the-art BLEU@4 and METEOR scores among PEFT methods and competitive performance with full fine-tuning using only 1.5% of the parameters. These results highlight Q-Adapter's strong trade-off between caption quality and parameter efficiency, supporting its scalability for video-language modeling.

1. Introduction

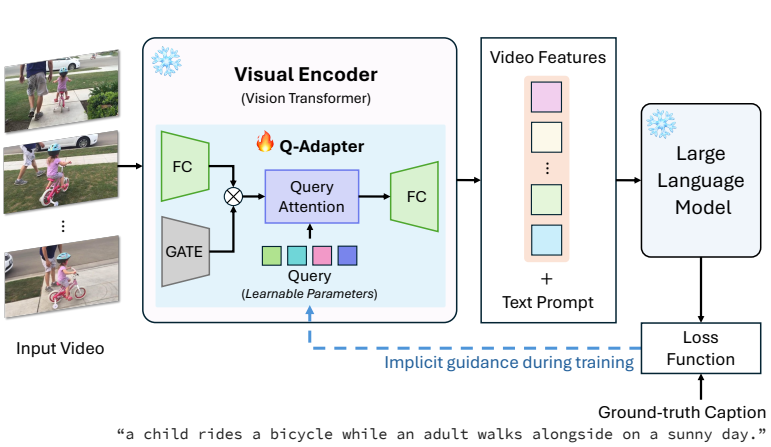
With the growing demand for advanced video understanding, the field of video captioning is rapidly developing. It aims to generate accurate and contextually relevant textual descriptions of video content. Early studies [21], [22] in video captioning primarily employed template-based methods, formulating the task as word prediction at predefined positions within fixed sentence structures. In recent years, the focus has shifted from word-level to sentence-level supervision, with increasing emphasis on end-to-end generation frameworks that align visual encoder features with semantic representations [17]. This alignment transforms video data into structured embeddings that language decoders can interpret effectively.

To improve cross-modal alignment, several studies [16], [25] have introduced additional supervisory signals to enhance the extraction of semantically relevant visual features. However, the limited size and diversity of publicly available video-caption datasets present significant challenges for effective training. To address this, pretraining based on contrastive learning for image-text alignment [10], [11], [20] has been widely adopted, with subsequent task-specific post-training applied to capture temporal dynamics in video [4], [15], [26], [28], [30], [34]. While these methods leverage alignment capabilities acquired through contrastive learning, they largely follow the conventional paradigm of "pre-training followed by fine-tuning," where full model tuning often leads to challenges such as catastrophic forgetting and suboptimal parameter efficiency [35].

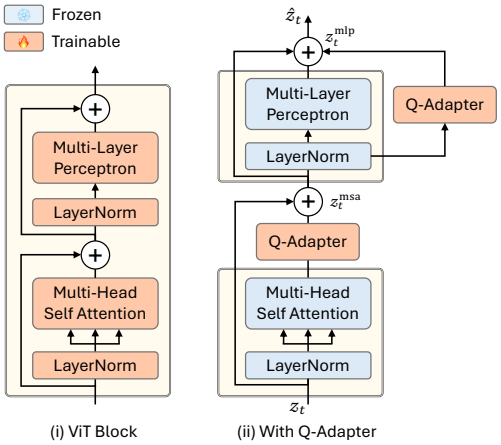
Recently, the development of Multimodal Large Language Models (MLLMs) has introduced new approaches for efficient fine-tuning in video-language tasks [7], [10], [12], [33], [36]. Methods such as VTimeLLM [7] employ Low-Rank Adaptation (LoRA) [6] to the language encoder with multi-stage training on curated question-answering datasets, showing promising results in dense video captioning. Other methods incorporate external textual knowledge via Retrieval-Augmented Generation (RAG) to enhance visual understanding [8], [9]. While these methods have demonstrated effectiveness, they often depend heavily on large external text datasets and the reasoning abilities of MLLMs, rather than focusing on improving how the model extracts and utilizes meaningful visual information from videos. Moreover, they tend to overlook the sparse and uneven distribution of informative content in video, which limits the quality of visual grounding essential for accurate caption generation. To address these limitations, this study introduces an adapter-based fine-tuning method into the visual encoder of MLLMs, aiming to enhance visual representation while maintaining parameter efficiency.

In this study, we present **Q**uery-**Adapter** (**Q-Adapter**),

[1] Graduate School of Informatics, Nagoya University
[2] Center for Artificial Intelligence, Mathematical and Data Science, Nagoya University
[a)] chenj@cs.is.i.nagoya-u.ac.jp
[b)] nguyent@cs.is.i.nagoya-u.ac.jp
[c)] taka-coma@acm.org
[d)] ide@i.nagoya-u.ac.jp



(a) Proposed Q-Adapter integrated into the Visual Encoder to extract caption-relevant features via query-guided attention and gated layer.



(b) Comparison between (i) the standard Vision Transformer (ViT) block and (ii) the integration with the proposed Q-Adapter.

Fig. 1: Overview of the video captioning framework and integration of Q-Adapter into the Visual Encoder

a novel visual adaptation for fine-tuning MLLMs in video captioning tasks. The proposed method introduces *learnable query tokens* into a lightweight adapter architecture. Unlike prior methods that rely heavily on external text sources or retrieval-based supervision, Q-Adapter leverages Language Modeling loss as implicit guidance, allowing the model to dynamically learn which visual features are most relevant for caption generation. By selectively injecting queries into the Vision Encoder, the proposed method enables more effective identification of sparse and semantically informative frames, without requiring explicit annotations or external corpora. The key contributions are summarized as follows:

- We propose **Q-Adapter**, a query-guided visual adapter for efficient fine-tuning of MLLMs in video captioning, integrating learnable query tokens and a gating mechanism into the Visual Encoder.
- Q-Adapter selectively extracts sparse and caption-relevant visual features without relying on external textual supervision, improving visual grounding under parameter constraints.
- On the MSR-VTT benchmark [29], Q-Adapter achieves state-of-the-art performance among PEFT methods using only 1.5% of the model parameters, indicating a favorable balance between efficiency and caption quality.

2. Q-Adapter-based Fine-tuning

Conventional full fine-tuning updates all the parameters of a pretrained model during training. In contrast, adapter fine-tuning [5] freezes the pretrained backbone and updates only a small set of newly introduced adapter parameters. Let $\boldsymbol{\phi}_w(\cdot)$ denote a pretrained neural network with parameters w, and let $\mathcal{D} = \{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^{N}$ be the training dataset, the optimization objectives of full and adapter fine-tuning can be formalized as:

$$\text{Full fine-tuning:} \quad \arg\min_{w} \mathcal{L}(\boldsymbol{\phi}_w(\boldsymbol{x}), \boldsymbol{y}), \tag{1}$$

$$\text{Adapter fine-tuning:} \quad \arg\min_{\theta} \mathcal{L}(\boldsymbol{\phi}_{w,\theta}(\boldsymbol{x}), \boldsymbol{y}), \tag{2}$$

where θ represents the adapter parameters and $\mathcal{L}$ denotes the task-specific loss function. In Equation 2, the backbone parameters w are frozen while only the adapter θ is updated.

In adapter fine-tuning, the placement of adapter modules is often determined by the characteristics of the downstream task. While adapters are typically inserted into intermediate layers of the network, there is no universally optimal configuration. Prior studies have shown that the effectiveness of adapter placement can vary significantly depending on the task objectives and data modalities [3], [5], [19], [23], [31]. This motivates the need for task-specific design choices when integrating adapters to fine-tune MLLMs.

2.1 Overview of the Proposed Framework

Figure 1(a) shows an overview of the proposed video captioning framework. Given an input video represented as a sequence of T frames $\mathcal{V} = \{\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_T\}$, the goal of video captioning is to generate a textual description $\boldsymbol{y} = \{y_1, y_2, \ldots, y_L\}$, where each y_i denotes a token in the output caption of length L. Each video frame $\boldsymbol{v}_t \in \mathbb{R}^{H\times W\times C}$, where H, W, and C denote the height, width, and number of channels of the input frame, respectively, is processed by a Visual Encoder to produce a frame-level representation $\boldsymbol{z}_t \in \mathbb{R}^{N\times d}$, where N is the number of visual tokens and d is the embedding dimension. These representations are then temporally aggregated into a unified video-level representation as:

$$\boldsymbol{z} = \mathcal{F}(\{\boldsymbol{z}_1, \boldsymbol{z}_2, \ldots, \boldsymbol{z}_T\}), \tag{3}$$

where $\mathcal{F}(\cdot)$ denotes a generic temporal fusion function (e.g., concatenation, pooling, or self-attention). The resulting video representation $\boldsymbol{z}$ is concatenated with a text prompt $\boldsymbol{p}$ and fed into a Text Decoder to generate the caption in an

autoregressive manner as:

$$y_i = \text{Decoder}(y_{<i}, [\boldsymbol{p}; \boldsymbol{z}]), \quad i = \{1, 2, \ldots, L\}, \tag{4}$$

where $y_{<i}$ denotes all previously generated tokens and $[\boldsymbol{p}; \boldsymbol{z}]$ represents the concatenation of the text prompt and the fused video features. The model is trained using the Cross Entropy loss, which minimizes the negative log-likelihood of the ground-truth caption as:

$$\mathcal{L} = -\sum_{i=1}^{L} \log p(y_i \mid y_{<i}, [\boldsymbol{p}; \boldsymbol{z}]), \tag{5}$$

The above formulation highlights that the effectiveness of video captioning largely depends on the quality of the video representation $\boldsymbol{z}$, which is obtained from the Visual Encoder. However, standard fine-tuning methods typically focus on the Language Decoder and struggle to adapt to the sparse and uneven distribution of informative content in videos. To enhance the quality of visual representations and enable efficient adaptation for downstream captioning tasks, we introduce Q-Adapter, which is integrated into the Visual Encoder. This adapter is designed as a lightweight visual module based on learnable query tokens that selectively extracts caption-relevant features without relying on additional text supervision.

2.2 Q-Adapter

Unlike conventional bottleneck adapters that rely solely on projection layers, Q-Adapter incorporates a learnable query-attention mechanism and a gating network to enhance its ability to extract sparse and semantically rich visual features.

2.2.1 Query-Attention Mechanism

For each video frame $t \in \{1, \ldots, T\}$, we denote the encoded feature map as $\boldsymbol{z}_t \in \mathbb{R}^{N \times d}$, where N is the number of visual tokens and d is the embedding dimension. To enhance the discriminative capacity of these features, Q-Adapter introduces a set of learnable query tokens $\boldsymbol{q} \in \mathbb{R}^{M \times d}$, where M is the number of queries. These tokens are designed to extract task-relevant information from $\boldsymbol{z}_t$ via cross-attention. We follow the standard scaled dot-product attention mechanism, where queries Q, keys K, and values V are defined as:

$$Q = \boldsymbol{q}\mathrm{W}^Q, \quad K = \boldsymbol{z}_t\mathrm{W}^K \; and \quad V = \boldsymbol{z}_t\mathrm{W}^V, \tag{6}$$

where $\mathrm{W}^Q, \mathrm{W}^K$, and $\mathrm{W}^V \in \mathbb{R}^{d \times d}$ are learned projection matrices. The query-guided attention output is then computed as:

$$\boldsymbol{o}_t = \text{LayerNorm}\left(\text{softmax}\left(\frac{QK^\top}{\sqrt{d}}\right)V\right), \tag{7}$$

where $\boldsymbol{o}_t \in \mathbb{R}^{M \times d}$ is the attended representation, capturing the most relevant information from $\boldsymbol{z}_t$ with respect to each query token. To preserve the spatial structure required for subsequent temporal fusion, the output is projected back to the original token length via a learned transformation $\Psi : \mathbb{R}^{M \times d} \to \mathbb{R}^{N \times d}$, followed by residual addition as:

$$\boldsymbol{z}_t^{\mathrm{q}} = \Psi(\boldsymbol{o}_t) + \boldsymbol{z}_t, \tag{8}$$

This formulation enables Q-Adapter to act as a content-aware refinement layer, selectively attending to semantically important regions while maintaining computational efficiency and structural alignment with the original token space.

2.2.2 Gated Feature Layer

To account for the potential distribution mismatch caused by freezing the pretrained encoder weights, we introduce a gating mechanism that adaptively modulates the input of the adapter. Specifically, we apply a Fully Connected (FC) layer to project the input feature $\boldsymbol{z}_t \in \mathbb{R}^{N \times d}$, followed by element-wise multiplication with a learnable gating layer. The gated output is computed as:

$$\boldsymbol{z}_t^{\mathrm{g}} = \text{Gate}(\boldsymbol{z}_t) \circ \text{FC}(\boldsymbol{z}_t), \tag{9}$$

where $\text{FC}(\cdot) : \mathbb{R}^{N \times d} \to \mathbb{R}^{N \times d'}$ is a dimensionality-reducing linear transformation, $\text{Gate}(\cdot)$ is a learnable gating layer implemented as a multi-layer perceptron, and $\circ$ denotes element-wise multiplication. The gated representation $\boldsymbol{z}_t^{\mathrm{g}}$ is used as the Key (K) and Value (V) inputs in the query-attention mechanism.

2.2.3 Integration into Visual Encoder

As illustrated in Figure 1(b), we integrate the proposed Q-Adapter into each block of the Visual Encoder to enable PEFT. Specifically, we apply Q-Adapter to each Vision Transformer (ViT) block: one for the Multi-head Self-Attention (MSA) layer and another for the Multi-Layer Perceptron (MLP) layer. This design preserves the original architecture while introducing a lightweight adaptation path that refines visual features without modifying pretrained backbone parameters. Let $\boldsymbol{z}_t \in \mathbb{R}^{N \times d}$ be the input token sequence to the block, the processing steps are defined as:

$$\boldsymbol{z}_t^{\mathrm{MSA}} = \boldsymbol{z}_t + \text{Q-Adapter}(\text{MSA}(\text{LayerNorm}(\boldsymbol{z}_t))), \tag{10}$$

$$\boldsymbol{z}_t^{\mathrm{MLP}} = \boldsymbol{z}_t^{\mathrm{MSA}} + \text{MLP}(\text{LayerNorm}(\boldsymbol{z}_t^{\mathrm{MSA}})) + \text{Q-Adapter}(\text{LayerNorm}(\boldsymbol{z}_t^{\mathrm{MSA}})), \tag{11}$$

$$\hat{\boldsymbol{z}}_t = \boldsymbol{z}_t^{\mathrm{MLP}}, \tag{12}$$

where z^{MSA} is the output sequence of the MSA layer, and z^{MLP} is the output sequence of the MLP layer. In this structure, only the parameters of the Q-Adapter are updated during training, while the original components of the Visual Encoder remain frozen. This selective tuning allows efficient adaptation to video captioning tasks with minimal additional parameters.

3. Experimental Evaluation

We evaluate the proposed Q-Adapter on the widely used MSR-VTT benchmark [29], which consists of 10,000 YouTube videos, each approximately 10 seconds long, annotated with a total of 200,000 captions. Following prior works [15], [28], we adopt the standard split of 9,000 videos for training and 1,000 for testing.

Table 1: Comparison of the proposed method with other methods on the MSR-VTT benchmark [29]. Best and second-best scores for each Fine-Tuning Type (FT Type) are highlighted with **bold** and underlined fonts, respectively.

Method	FT Type	FT Ratio ↓	BLEU@4 ↑	METEOR ↑	ROUGE-L ↑	CIDEr ↑
MV-GPT [24]	Full	100%	48.90	**38.70**	64.00	60.00
HiTeA [32]			49.20	30.70	65.00	65.10
mPLUG-2_{Base} [28]			52.20	32.10	66.90	72.40
mPLUG-2 [28]			**57.80**	34.90	**70.10**	**80.30**
Qwen2.5-VL (Zero-shot) [1]	—	—	5.27	20.47	35.32	0.26
Qwen2.5-VL [1] + Adapter [5]	PEFT	2.1%	50.89	63.02	**63.93**	**66.45**
Qwen2.5-VL [1] + LoRA(Attn) [6]		**0.8%**	45.01	61.04	61.79	62.55
Qwen2.5-VL [1] + LoRA(MLP) [6]		1.5%	46.03	61.87	62.26	63.10
Qwen2.5-VL [1] + LoRA(Attn&MLP) [6]		2.3%	45.63	61.49	61.98	62.36
Qwen2.5-VL [1] + Q-Adapter (Proposed)		1.5%	**51.24**	**63.49**	62.52	65.70

3.1 Experimental Settings

3.1.1 Model & Hyperparameters

We adopt Qwen2.5-VL-3B-Instruct[*1] as the backbone for ViT and the Text Decoder (Qwen2.5 LM). The input resolution of each video frame is standardized to 224×224 pixels. Following Xu et al. [28], we randomly sample 16 frames per video during training and apply uniform sampling during inference to ensure stability and reproducibility. All experiments are trained for 10 epochs with a batch size of 8 using AdamW [14] with a learning rate of 2×10^{-4}, and a linear scheduler with a warmup ratio of 0.1.

3.1.2 Evaluation Metrics

To fairly evaluate the performance of PEFT methods, we define the Fine-Tuned Ratio (FT Ratio) as follows, highlighting the distinction between the PEFT approach and conventional full fine-tuning as:

$$\text{FT Ratio} = \frac{\text{Number of trainable parameters}}{\text{Total number of model parameters}}. \quad (13)$$

All methods are evaluated on the video captioning task using four standard metrics: BLEU@4 [18], METEOR [2], ROUGE-L [13], and CIDEr [27]. These metrics measure the quality of generated captions based on their similarity to ground-truth references.

3.1.3 Comparison Methods

We compare the proposed method with representative PEFT methods under a unified evaluation setting.

- **Adapter** [5]: Standard adapter layers are inserted after the MSA and MLP components of each ViT block, with a gating mechanism to enhance stability.

- **LoRA** [6]: Learnable low-rank matrices are injected in parallel with the weights. We evaluate the following three integration strategies for each ViT block:
  - LoRA(Attn): Applied to the projection matrices Q, K, and V within the MSA layers.
  - LoRA(MLP): Applied to the feedforward network following the MSA layers.
  - LoRA(Attn&MLP): Applied to all trainable components within each ViT block.

[*1] https://huggingface.co/Qwen/Qwen2.5-VL-3B-Instruct/ (Accessed: 2025-06-05)

3.2 Results

Table 1 presents a comparison between the proposed Q-Adapter and existing fine-tuning methods. Within the PEFT methods, the proposed Q-Adapter achieved the best scores on BLEU@4, METEOR, and a low FT Ratio (1.5%), while also securing second-best scores on ROUGE-L and CIDEr. Notably, the Adapter required more trainable parameters (2.1% FT Ratio) to achieve slightly better scores on ROUGE-L and CIDEr. Similarly, LoRA-based methods, despite tuning 2.3% of parameters, consistently underperformed in both accuracy and efficiency. These results underscore the strength of Q-Adapter in achieving a better trade-off between parameter efficiency and caption quality, driven by its ability to selectively extract task-relevant visual features with minimal overhead.

When compared to full fine-tuning methods, Q-Adapter demonstrated competitive performance despite tuning only a small fraction of the model parameters. It outperformed both MV-GPT [24] and HiTeA [32] across BLEU@4, METEOR, and CIDEr metrics. Although mPLUG-2 [28] achieved higher scores, the results show that Q-Adapter offers a strong balance between parameter efficiency and caption quality, making it an appealing alternative for fine-tuning under limited resource constraints.

4. Conclusion

We proposed Q-Adapter, a lightweight visual fine-tuning framework for video captioning that integrates query-guided attention and gated modulation into the Visual Encoder of MLLMs. The proposed method performed well on the MSR-VTT benchmark [29], surpassing existing PEFT methods with minimal parameter updates and showed competitive results against full fine-tuning.

In future work, we plan to explore enhanced query supervision and apply Q-Adapter to a broader range of video captioning benchmarks and video-language understanding tasks.

Acknowledgments

This work was partly supported by Japan Society for the Promotion of Science (JSPS) KAKENHI 23K24868 and 25K00161.

References

[1] Bai, S., Chen, K., Liu, X., Wang, J., Ge, W., Song, S., Dang, K., Wang, P., Wang, S., Tang, J., Zhong, H., Zhu, Y., Yang, M., Li, Z., Wan, J., Wang, P., Ding, W., Fu, Z., Xu, Y., Ye, J., Zhang, X., Xie, T., Cheng, Z., Zhang, H., Yang, Z., Xu, H. and Lin, J.: Qwen2.5-VL Technical Report, *Computing Research Repository arXiv Preprint,* arXiv:2502.13923 (2025).

[2] Banerjee, S. and Lavie, A.: METEOR: An automatic metric for MT evaluation with improved correlation with human judgments, *Proc. ACL2005 Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pp. 65–72 (2005).

[3] Chen, S., Ge, C., Tong, Z., Wang, J., Song, Y., Wang, J. and Luo, P.: AdaptFormer: Adapting vision Transformers for scalable visual recognition, *Advances in Neural Information Processing Systems*, Vol. 35, pp. 16664–16678 (2022).

[4] Chen, S., Li, H., Wang, Q., Zhao, Z., Sun, M., Zhu, X. and Liu, J.: VAST: A Vision-Audio-Subtitle-Text omni-modality foundation model and dataset, *Advances in Neural Information Processing Systems*, Vol. 36, pp. 72842–72866 (2023).

[5] Houlsby, N., Giurgiu, A., Jastrzebski, S., Morrone, B., De Laroussilhe, Q., Gesmundo, A., Attariyan, M. and Gelly, S.: Parameter-efficient transfer learning for NLP, *Proc. 36th International Conference on Machine Learning*, pp. 2790–2799 (2019).

[6] Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L. and Chen, W.: LoRA: Low-Rank Adaptation of large language models, *Proc. 10th International Conference on Learning Representations*, pp. 3–16 (2022).

[7] Huang, B., Wang, X., Chen, H., Song, Z. and Zhu, W.: VTimeLLM: Empower LLM to grasp video moments, *Proc. 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14271–14280 (2024).

[8] Kim, M., Kim, H. B., Moon, J., Choi, J. and Kim, S. T.: Do you remember? Dense video captioning with cross-modal memory retrieval, *Proc. 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13894–13904 (2024).

[9] Li, B., Yuan, C., Hu, W., Deng, Y., Shan, Y., Qi, Z. and Zhang, Z.: Open-book video captioning with retrieve-copy-generate network, *Proc. 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9837–9846 (2021).

[10] Li, J., Li, D., Savarese, S. and Hoi, S.: BLIP-2: Bootstrapping Language-Image Pre-training with frozen image encoders and large language models, *Proc. 40th International Conference on Machine Learning*, pp. 19730–19742 (2023).

[11] Li, J., Selvaraju, R., Gotmare, A., Joty, S., Xiong, C. and Hoi, S. C. H.: Align before fuse: Vision and language representation learning with momentum distillation, *Advances in Neural Information Processing Systems*, Vol. 34, pp. 9694–9705 (2021).

[12] Li, K., He, Y., Wang, Y., Li, Y., Wang, W., Luo, P., Wang, Y., Wang, L. and Qiao, Y.: VideoChat: Chat-centric video understanding, *Computing Research Repository arXiv Preprint,* arXiv:2305.06355 (2023).

[13] Lin, C.-Y.: ROUGE: A package for automatic evaluation of summaries, *Proc. ACL2004 Workshop on Text Summarization Branches Out*, pp. 74–81 (2004).

[14] Loshchilov, I. and Hutter, F.: Decoupled weight decay regularization, *Computing Research Repository arXiv Preprint,* arXiv:1711.05101 (2017).

[15] Luo, H., Ji, L., Zhong, M., Chen, Y., Lei, W., Duan, N. and Li, T.: CLIP4Clip: An empirical study of CLIP for end to end video Clip retrieval and captioning, *Neurocomputing*, Vol. 508, pp. 293–304 (2022).

[16] Pan, B., Cai, H., Huang, D. A., Lee, K. H., Gaidon, A., Adeli, E. and Niebles, J. C.: Spatio-temporal graph for video captioning with knowledge distillation, *Proc. 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10867–10876 (2020).

[17] Pan, Y., Mei, T., Yao, T., Li, H. and Rui, Y.: Jointly modeling embedding and translation to bridge video and language, *Proc. 2016 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4594–4602 (2016).

[18] Papineni, K., Roukos, S., Ward, T. and Zhu, W.-J.: Bleu: A method for automatic evaluation of machine translation, *Proc. 40th Annual Meeting of the Association for Computational Linguistics*, pp. 311–318 (2002).

[19] Pfeiffer, J., Kamath, A., Rücklé, A., Cho, K. and Gurevych, I.: AdapterFusion: Non-destructive task composition for transfer learning, *Proc. 16th Conference of the European Chapter of the Association for Computational Linguistics*, pp. 487–503 (2021).

[20] Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G. and Sutskever, I.: Learning transferable visual models from natural language supervision, *Proc. 38th International Conference on Machine Learning*, pp. 8748–8763 (2021).

[21] Rohrbach, A., Rohrbach, M., Tandon, N. and Schiele, B.: A dataset for movie description, *Proc. 2015 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3202–3212 (2015).

[22] Rohrbach, M., Qiu, W., Titov, I., Thater, S., Pinkal, M. and Schiele, B.: Translating video content to natural language descriptions, *Proc. 14th IEEE International Conference on Computer Vision*, pp. 433–440 (2013).

[23] Rücklé, A., Geigle, G., Glockner, M., Beck, T., Pfeiffer, J., Reimers, N. and Gurevych, I.: AdapterDrop: On the efficiency of adapters in Transformers, *Proc. 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 7930–7946 (2021).

[24] Seo, P. H., Nagrani, A., Arnab, A. and Schmid, C.: End-to-end generative pretraining for multimodal video captioning, *Proc. 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 17959–17968 (2022).

[25] Shi, B., Ji, L., Niu, Z., Duan, N., Zhou, M. and Chen, X.: Learning semantic concepts and temporal alignment for narrated video procedural captioning, *Proc. 28th ACM International Conference on Multimedia*, pp. 4355–4363 (2020).

[26] Tang, M., Wang, Z., Liu, Z., Rao, F., Li, D. and Li, X.: CLIP4Caption: CLIP for video Caption, *Proc. 29th ACM International Conference on Multimedia*, pp. 4858–4862 (2021).

[27] Vedantam, R., Lawrence Zitnick, C. and Parikh, D.: CIDEr: Consensus-based Image Description Evaluation, *Proc. 2015 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4566–4575 (2015).

[28] Xu, H., Ye, Q., Yan, M., Shi, Y., Ye, J., Xu, Y., Li, C., Bi, B., Qian, Q., Wang, W., Xu, G., Zhang, J., Huang, S., Huang, F. and Zhou, J.: mPLUG-2: A modularized multi-modal foundation model across text, image and video, *Proc. 40th International Conference on Machine Learning*, pp. 38728–38748 (2023).

[29] Xu, J., Mei, T., Yao, T. and Rui, Y.: MSR-VTT: A large video description dataset for bridging video and language, *Proc. 2016 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5288–5296 (2016).

[30] Yang, A., Nagrani, A., Seo, P. H., Miech, A., Pont-Tuset, J., Laptev, I., Sivic, J. and Schmid, C.: Vid2Seq: Large-scale pretraining of a visual language model for dense video captioning, *Proc. 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10714–10726 (2023).

[31] Yang, T., Zhu, Y., Xie, Y., Zhang, A., Chen, C. and Li, M.: AIM: Adapting Image Models for efficient video action recognition, *Proc. 11th International Conference on Learning Representations*, pp. 1–14 (2023).

[32] Ye, Q., Xu, G., Yan, M., Xu, H., Qian, Q., Zhang, J. and Huang, F.: HiTeA: Hierarchical Temporal-Aware video-language pre-training, *Proc. 19th IEEE/CVF International Conference on Computer Vision*, pp. 15405–15416 (2023).

[33] Yin, S., Fu, C., Zhao, S., Li, K., Sun, X., Xu, T. and Chen, E.: A survey on multimodal large language models, *Computing Research Repository arXiv Preprint,* arXiv:2306.13549 (2023).

[34] Yu, K., Zhang, Z., Hu, F., Storks, S. and Chai, J.: Eliciting in-context learning in vision-language models for videos through curated data distributional properties, *Proc. 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 20416–20431 (2024).

[35] Zhai, Y., Tong, S., Li, X., Cai, M., Qu, Q., Lee, Y. J. and Ma, Y.: Investigating the catastrophic forgetting in multimodal large language models, *Computing Research Repository arXiv Preprint,* arXiv:2309.10313 (2023).

[36] Zhang, H., Li, X. and Bing, L.: Video-LLaMA: An instruction-tuned audio-visual language model for video understanding, *Proc. 2023 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pp. 543–553 (2023).