Zero-Shot and Few-Shot Video Question Answering with Multi-Modal Prompts
Supplementary Material

Deniz Engin[1] Yannis Avrithis[2]

[1]Inria, Univ Rennes, CNRS, IRISA
[2]Institute of Advanced Research in Artificial Intelligence (IARAI)

A. Experimental Setup

A.1. Datasets

Pretraining We use WebVid2M [3] for pretraining, consisting of 2.5M video-caption pairs scraped from the internet. The domain is open and the captions are manually generated. The average video duration is 18 seconds and the average caption word count is 12.

Downstream tasks Downstream dataset statistics are given in Table 4. Following [57], we use 1% of the training data for fine-tuning in the few-shot setting.

MSRVTT-QA [53] is an extension of MSR-VTT [54], where question-answer pairs are automatically generated from video descriptions. MSVD-QA [53] is based on MSVD [7] and question-answers pairs are automatically generated as in MSRVTT-QA. ActivityNet-QA [58] is derived from ActivityNet [6]. The average video duration is 180s. TGIF-QA [21] comprises several tasks, including FRAME-QA, where the question can be answered from one of the frames in a GIF. In this work, TGIF-QA refers only to Frame-QA.

Dataset	Videos	QA Pairs			
		Train	Val	Test	Total
MSRVTT-QA. [53]	10k	159k	12k	73k	244k
MSVD-QA. [53]	2k	31k	6.5k	13k	50.5k
ActivityNet-QA [58]	5.8k	32k	18k	8k	58k
TGIF-QA [21]	40k	39k	–	13k	53k

Table 4: Downstream dataset statistics.

A.2. Implementation Details

Text prompt parametrization Instead of defining text prompts as parameters directly, we discuss here an alternative parametrization using projections. We first generate a sequence of input prompts $P^i \in \mathbb{R}^{D' \times N}$ and then we project it as follows:

$$P^t := W P^i \in \mathbb{R}^{2CD \times N}, \tag{8}$$

where $W \in \mathbb{R}^{2CD \times D'}$, C is the number of layers of the language model f and D its embedding dimension. Then, P^t can be reshaped as a $2 \times C \times D \times N$ tensor, representing one pair of sequences $P_K, P_V \in \mathbb{R}^{D \times N}$ for every layer of f. After training, the input sequence P^i and projection matrix W are discarded and we only keep P^t. This allows us to fine-tune fewer parameters at downstream tasks, which is beneficial when data is limited.

Architecture details The *frozen video encoder* is CLIP ViT-L/14 [10, 44], trained with contrastive loss on 400M image-text pairs. We uniformly sample $T = 10$ frames located at least 1 second apart and each frame is resized to 224×224 pixels; if the video is shorter than 10 seconds, we zero-pad up to $T = 10$ frames. The encoder then extracts one feature vector per frame of the dimension of 768, followed by a linear projection to $D = 1536$ dimensions.

The *visual mapping network* has $L = 2$ layers, each with a cross-attention and a self-attention, having 8 heads and embedding dimension $D = 1536$. We use $M = 10$ learnable visual prompt vectors of dimension $D = 1536$.

The *text tokenizer* is based on SentencePiece [26] with a vocabulary U of size 128k.

The *frozen language model* is DeBERTa-V2-XLarge [17], trained using MLM on 160G text data, following [57]. The model has $C = 24$ layers, 24 attention heads, and embedding dimension $D = 1536$, resulting in 900M parameters.

For the *adapter layers* [18], we set $d = D/8 = 192$ by following [57]. For *text prompts*, we use $N = 10$ learnable text prompt vectors, $D' = D/8 = 192$, and $C = 24$.

Downstream input design We limit the length of text sequences to $S = 256$ tokens for pretraining and zero-shot experiments and $S = 128$ tokens for downstream experiments. We adopt the input design of [57] as follows: ”[CLS] Question: <Question>? Answer: [MASK].

Subtitles: <Subtitles> [SEP]". Subtitles are optional and if available, their token sequence X^s is incorporated into the input. In this case, the text input sequence becomes $X^t = (X^q, X^a, X^s)$.

Answer vocabulary The answer vocabulary U is constructed by selecting the top 1k most frequent answers from the training set for the zero-shot setting, following [57, 60]. Another vocabulary is formed by including answers that occur at least twice in the training set for the few-shot setting, as defined in [57]. Questions with answers outside the vocabulary are excluded from the training process and are assessed as incorrect during evaluation. To report results for the few-shot setting, we choose the vocabulary that yields the best performance on the validation set.

Answer embedding The classifier head of the frozen language model includes more tokens than required for downstream training. To address this, by following [57], we define a task-specific classification head by keeping the weights of the pretrained head associated with the answer vocabulary. At inference, we provide one mask token at the input, regardless of the ground truth answer length, and we obtain one output logit vector. For multi-token answers, we take the average of the logits corresponding to the ground truth words from the vocabulary.

Training settings We use the Adam optimizer [25] with $\beta = (0.9, 0.95)$ in all experiments. We decay the learning rate using a linear schedule with the warm-up in the first 10% of the iterations. We use dropout with probability 0.1 in the language model, adapter layers, text prompts, and visual mapping network. We adopt automatic mixed precision training for all experiments.

We *pretrain* for 10 epochs on WebVid2M with a batch size of 128 on 8 NVIDIA Tesla V100 GPUs, amounting to 20 hours total training time. The base learning rate is 2×10^{-5} and the learning rate for visual and text prompts is separately set to 10^{-3}.

For *fine-tuning* on each downstream dataset, we train for 20 epochs with a batch size of 32 on 4 NVIDIA Tesla V100 GPUs. The base learning rate is searched over 5 values in the interval $[10^{-5}, 5 \times 10^{-5}]$, while the learning rate for visual and text prompts is kept at 10^{-3}. For *prompt-only fine-tuning*, the base learning rate is searched over 3 values in the interval $[10^{-3}, 3 \times 10^{-3}]$.

B. More ablations

Prompt length Figure 2 shows the effect of the number of prompts on few-shot performance, referring to both visual (M) and text (N) prompts, *i.e.*, $M = N$. Because the space and time complexity of the model is quadratic in the number of prompts, we limit this number to 50. We find that accuracy is consistently best on all downstream benchmarks for $M = N = 10$ prompts, which we choose as default.

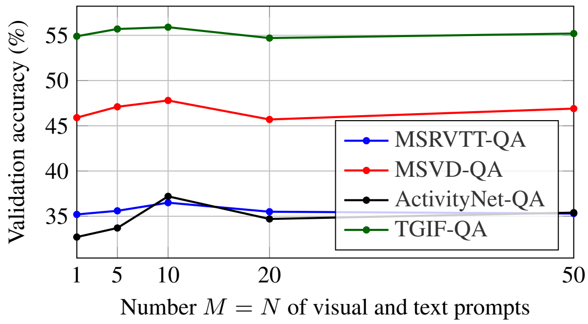

Figure 2: Few-shot top-1 validation accuracy *vs.* number $M = N$ of *visual and text prompts* for different downstream datasets, using 1% of training data.

VPN Layers	MSRVTT-QA	MSVD-QA	ANet-QA	TGIF-QA
1	36.0	47.0	36.1	55.9
2	**36.5**	**47.8**	**37.2**	**55.9**

Table 5: Effect of number L of layers of our visual mapping network on few-shot top-1 validation accuracy, using 1% of training data. VPN: Visual Mapping Network. ANet-QA: ActivityNet-QA.

Reparam	MSRVTT-QA	MSVD-QA	ANet-QA	TGIF-QA
	35.6	47.4	34.0	55.1
✓	**36.5**	**47.8**	**37.2**	**55.9**

Table 6: Effect of reparametrization of text prompts on few-shot top-1 validation accuracy, using 1% of training data. Reparam: Reparametrization. ANet-QA: ActivityNet-QA.

Number of layers of visual mapping network Table 5 shows the effect of the number L of layers of our visual mapping network on few-shot performance. We only experiment with up to 2 layers to avoid an excessive number of parameters and complexity of our model. We find that $L = 2$ works best, which we choose as default.

Reparametrization of text prompts In Table 6, we investigate the impact of the reparametrization of text prompts, as discussed in Subsection A.2, on few-shot performance. We find that reparametrization consistently improves performance on all downstream benchmarks. Even though the number of trainable parameters increases from 87M to 101M during pretraining and fine-tuning, we do not need to store the additional parameters at inference.

Handcrafted prompts We explore the use of handcrafted prompts in the input text. In Table 7 and Table 8, we con-

#	Input Design	MSRVTT -QA	MSVD -QA	ANet -QA	TGIF -QA
1	"[CLS] <Question>? [MASK]. <Subtitles> [SEP]"	13.2	30.2	19.8	29.8
2	"[CLS] **Answer the question:** <Question>? [MASK]. <Subtitles> [SEP]"	7.8	22.3	14.3	35.3
3	"[CLS] <Question>? **Answer:** [MASK]. <Subtitles> [SEP]"	17.7	37.2	**25.8**	45.1
4	"[CLS] **Question:** <Question>? **Answer:** [MASK]. **Subtitles:** <Subtitles> [SEP]"	**18.0**	**38.2**	24.9	**45.5**

Table 7: Effect of handcrafted prompt placement on *zero-shot* top-1 validation accuracy. ANet-QA: ActivityNet-QA.

#	Input Design	MSRVTT -QA	MSVD -QA	ANet -QA	TGIF -QA
1	"[CLS] <Question>? [MASK]. <Subtitles> [SEP]"	36.3	47.0	35.8	55.8
2	"[CLS] **Answer the question**: <Question>? [MASK]. <Subtitles> [SEP]"	36.3	46.8	35.1	55.8
3	"[CLS] <Question>? **Answer:** [MASK]. <Subtitles> [SEP]"	**36.5**	47.1	35.9	55.8
4	"[CLS] **Question:** <Question>? **Answer:** [MASK]. **Subtitles:** <Subtitles> [SEP]"	**36.5**	**47.8**	**37.2**	**55.9**

Table 8: Effect of handcrafted prompt placement on *few-shot* top-1 validation accuracy, using 1% of training data. ANet-QA: ActivityNet-QA.

Method	Sub	#Training Img	Vid	VQA	MSRVTT-QA	MSVD-QA	ANet-QA	TGIF-QA
CLIP [44]		400M	-		2.1	7.2	1.2	3.6
Reserve [59]	✓	-	20M		5.8	-	-	-
Lavender [34]		3M	2.5M		4.5	11.6	-	16.7
Flamingo-3B [1]		2.3B	27M		11.0	27.5	-	-
Flamingo-9B [1]		2.3B	27M		13.7	30.2	-	-
Flamingo [1]		2.3B	27M		17.4	35.6	-	-
FrozenBiLM [57]	✓	-	10M		16.7	33.8	**25.9**	41.9
Just Ask [55]		69M	-	✓	2.9	7.5	12.2	-
Just Ask [56]		69M	3M	✓	5.6	13.5	12.3	-
BLIP [32]		129M	-	✓	19.2	35.2	-	-
ViTiS (Ours)		-	2.5M		**18.2**	**36.2**	25.0	**45.5**
ViTiS (Ours)	✓	-	2.5M		18.1	36.1	25.5	**45.5**

Table 9: Extended version of Table 2, providing more results on *zero-shot VideoQA* top-1 accuracy on test sets, except TGIF-QA on the validation set. Number of pretraining data: image-text/video-text pairs. VQA: visual question answer pairs. Sub: subtitle input. ANet-QA: ActivityNet-QA. CLIP: CLIP ViT-L/14. Flamingo: Flamingo-80B. We gray out methods trained on VQA pairs, which are not directly comparable.

Method	#Shot	#Pre-Training Img	Vid	#Param	MSRVTT-QA	MSVD-QA	ANet-QA	TGIF-QA
Flamingo-3B [1]	32	2.3B	27M	1.4B	25.6	42.6	–	–
Flamingo-9B [1]	32	2.3B	27M	1.8B	29.4	47.2	–	–
Flamingo-80B [1]	32	2.3B	27M	10B	31.0	52.3	–	–
ViTiS (Ours)	32	–	2.5M	101M	27.0±1.0	41.9±0.8	28.7±1.3	52.2±1.2

Table 10: *Few-shot VideoQA in-context learning*. Mean and standard deviation of top-1 accuracy on test sets, except TGIF-QA on the validation set, over 10 32-shot tasks drawn at random. Only our model involves parameter updates; we fine-tune 0.75M params. Number of pretraining data: image-text/video-text pairs. ANet-QA: ActivityNet-QA.

sider four different input designs for zero-shot and few-shot settings, respectively: (i) no handcrafted prompts, (ii) placed before the question, (iii) placed just before the [MASK] token (answer), and (iv) placed just before the question, answer and subtitles.

In *zero-shot*, handcrafted prompts are beneficial due to the absence of task-specific learning for downstream tasks. As shown in Table 7, the absence of handcrafted prompts drastically reduces the performance (row 1), highlighting their necessity. Moreover, the position of the handcrafted prompt has a significant impact on the performance. More specifically, the location of the "Answer" prompt affects the results by a large margin (row 2→3), even leading to worse performance than the absence of handcrafted prompts (row 1→2). The presence of an "Answer" prompt just before the [MASK] token yields better performance in two input designs (rows 3 & 4).

Although the impact of using handcrafted text prompts is relatively small in *few-shot* experiments compared to zero-shot experiments, they still improve enhances, particularly on MSRVTT-QA and TGIF-QA datasets, as shown in Table 8. Placing handcrafted prompts at the beginning (row 2), as is the case for learnable text prompts, leads to lower performance. The best performance is achieved when handcrafted prompts are placed just before the question, answer, and subtitles (row 4). Therefore, we choose to place handcrafted prompts according to row 4 for both settings.

By contrast, *learable prompts* are all placed at the beginning. We empirically observe that other choices, *e.g.* placing half at the beginning of the input and half just before the [MASK] token, are inferior.

C. Additonal Results

Zero-shot results Table 9 is an extended version of Table 2, providing a comparison with state-of-the-art methods for zero-shot VideoQA. It includes results for additional versions of Flamingo [1], which our method outperforms all. It also includes two more methods that are not directly comparable with our zero-shot settings. In particular, BLIP [32] is pretrained on the VQA dataset [15], which is not directly comparable as our setting does not involve training on QA pairs. Similarly, Just Ask [55, 56] leverages automatically generated visual question answering datasets; although these datasets are not annotated by humans, the model is still trained on the specific task.

Few-shot results An alternative approach for few-shot VideoQA is *in-context learning* [1], using few, *e.g.* 32, labeled examples. To compare, we draw 10 tasks of 32 examples at random from 1% of training data of each downstream dataset, we fine-tune the prompt vectors, that is, 0.75M parameters, on each task for 5 epochs and report mean and standard deviation. This can be considered as *test-time prompt tuning* [47] using task-specific annotated data.

Table 10 shows the results of few-shot in-context learning. Flamingo [1] uses a frozen auto-regressive language model with trainable cross-attention layers that incorporate vision and language input, trained on an extreme-scale dataset. The Flamingo-3B, Flamingo-9B, and Flamingo-80B have 1.4B, 1.8B, and 10B learned parameters, respectively, in addition to the frozen language model. By contrast, our method uses a lighter frozen language model and lighter adaptation modules, resulting in only 101M parameters to learn, and our training data is a relatively small amount of video-text pairs. Despite this, our method outperforms Flamingo-3B [1] on MSRVTT-QA and is on par with MSVD-QA.

References

[1] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. In *Proc. NeurIPS*, 2022. 1, 4, 3

[2] Hyojin Bahng, Ali Jahanian, Swami Sankaranarayanan, and Phillip Isola. Exploring visual prompts for adapting large-scale models. *arXiv preprint arXiv:2203.17274*, 2022. 2

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

[4] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. In *Proc. NeurIPS*, 2020. 2

[5] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. In *Proc. NeurIPS*, 2020. 4

[6] Fabian Caba Heilbron, Victor Escorcia, Bernard Ghanem, and Juan Carlos Niebles. ActivityNet: A large-scale video benchmark for human activity understanding. In *Proc. CVPR*, 2015. 1

[7] David Chen and William Dolan. Collecting highly parallel data for paraphrase evaluation. In *Proc. ACL*, 2011. 1

[8] Feng Cheng, Xizi Wang, Jie Lei, David Crandall, Mohit Bansal, and Gedas Bertasius. VindLU: A recipe for effective video-and-language pretraining. In *Proc. CVPR*, 2023. 1

[9] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proc. NAACL*, 2019. 1

[10] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *Proc. ICLR*, 2021. 1, 3

[11] Constantin Eichenberg, Sidney Black, Samuel Weinbach, Letitia Parcalabescu, and Anette Frank. MAGMA–multimodal augmentation of generative models through adapter-based finetuning. In *Proc. Findings of EMNLP*, 2022. 2

[12] Tsu-Jui Fu, Linjie Li, Zhe Gan, Kevin Lin, William Yang Wang, Lijuan Wang, and Zicheng Liu. Violet: End-to-end video-language transformers with masked visual-token modeling. *arXiv preprint arXiv:2111.12681*, 2021. 1

[13] Tsu-Jui Fu, Linjie Li, Zhe Gan, Kevin Lin, William Yang Wang, Lijuan Wang, and Zicheng Liu. An empirical study of end-to-end video-language transformers with masked visual modeling. In *Proc. CVPR*, 2023. 1

[14] Noa Garcia, Mayu Otani, Chenhui Chu, and Yuta Nakashima. KnowIT VQA: Answering knowledge-based questions about videos. In *Proc. AAAI*, 2020. 1

[15] Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the V in VQA matter: Elevating the role of image understanding in Visual Question Answering. In *Proc. CVPR*, 2017. 4

[16] Tengda Han, Max Bain, Arsha Nagrani, Gül Varol, Weidi Xie, and Andrew Zisserman. AutoAD: Movie description in context. In *Proc. CVPR*, 2023. 1

[17] Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. DeBERTa: Decoding-enhanced bert with disentangled attention. In *Proc. ICLR*, 2021. 1, 4

[18] Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for NLP. In *Proc. ICML*, 2019. 1, 2

[19] Jingjia Huang, Yinan Li, Jiashi Feng, Xinglong Wu, Xiaoshuai Sun, and Rongrong Ji. Clover: Towards a unified video-language alignment and fusion model. In *Proc. CVPR*, 2023. 1

[20] Andrew Jaegle, Felix Gimeno, Andy Brock, Oriol Vinyals, Andrew Zisserman, and Joao Carreira. Perceiver: General perception with iterative attention. In *Proc. ICML*, 2021. 1, 2

[21] Yunseok Jang, Yale Song, Youngjae Yu, Youngjin Kim, and Gunhee Kim. Tgif-QA: Toward spatio-temporal reasoning in visual question answering. In *Proc. CVPR*, 2017. 1, 3

[22] Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. Visual prompt tuning. In *Proc. ECCV*, 2022. 2

[23] Chen Ju, Tengda Han, Kunhao Zheng, Ya Zhang, and Weidi Xie. Prompting visual-language models for efficient video understanding. In *Proc. ECCV*, 2022. 2

[24] Muhammad Uzair Khattak, Hanoona Rasheed, Muhammad Maaz, Salman Khan, and Fahad Shahbaz Khan. MaPLe: Multi-modal prompt learning. In *Proc. CVPR*, 2023. 2

[25] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *Proc. ICLR*, 2015. 2

[26] Taku Kudo and John Richardson. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proc. EMNLP: System Demonstrations*, 2018. 1

[27] Yi-Lun Lee, Yi-Hsuan Tsai, Wei-Chen Chiu, and Chen-Yu Lee. Multimodal prompting with missing modalities for visual recognition. In *Proc. CVPR*, 2023. 2

[28] 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 *Proc. CVPR*, 2021. 1

[29] Jie Lei, Licheng Yu, Mohit Bansal, and Tamara Berg. TVQA: Localized, compositional video question answering. In *Proc. EMNLP*, 2018. 1

[30] Brian Lester, Rami Al-Rfou, and Noah Constant. The power of scale for parameter-efficient prompt tuning. In *Proc. EMNLP*, 2021. 2

[31] Dongxu Li, Junnan Li, Hongdong Li, Juan Carlos Niebles, and Steven CH Hoi. Align and prompt: Video-and-language pre-training with entity prompts. In *Proc. CVPR*, 2022. 1

[32] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. BLIP: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *Proc. ICML*, 2022. 1, 3, 4

[33] Linjie Li, Yen-Chun Chen, Yu Cheng, Zhe Gan, Licheng Yu, and Jingjing Liu. HERO: Hierarchical encoder for video+ language omni-representation pre-training. In *Proc. EMNLP*, 2020. 1

[34] Linjie Li, Zhe Gan, Kevin Lin, Chung-Ching Lin, Zicheng Liu, Ce Liu, and Lijuan Wang. Lavender: Unifying video-language understanding as masked language modeling. In *Proc. CVPR*, 2023. 1, 4, 3

[35] Xiang Lisa Li and Percy Liang. Prefix-Tuning: Optimizing continuous prompts for generation. In *Proc. ACL*, 2021. 1, 2

[36] Xiao Liu, Kaixuan Ji, Yicheng Fu, Weng Tam, Zhengxiao Du, Zhilin Yang, and Jie Tang. P-tuning: Prompt tuning can be comparable to fine-tuning across scales and tasks". In *Proc. ACL*, 2022. 1, 2

[37] Xiao Liu, Yanan Zheng, Zhengxiao Du, Ming Ding, Yujie Qian, Zhilin Yang, and Jie Tang. GPT understands, too. *arXiv preprint arXiv:2103.10385*, 2021. 1, 2

[38] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. RoBERTa: A robustly optimized BERT pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019. 1

[39] Ze Liu, Jia Ning, Yue Cao, Yixuan Wei, Zheng Zhang, Stephen Lin, and Han Hu. Video swin transformer. In *Proc. CVPR*, 2022. 1

[40] Rabeeh Karimi Mahabadi, Luke Zettlemoyer, James Henderson, Lambert Mathias, Marzieh Saeidi, Veselin Stoyanov, and Majid Yazdani. PERFECT: prompt-free and efficient few-shot learning with language models. In *Proc. ACL*, 2022. 2

[41] 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 *Proc. ICCV*, 2019. 1

[42] Ron Mokady, Amir Hertz, and Amit H Bermano. ClipCap: CLIP prefix for image captioning. *arXiv preprint arXiv:2111.09734*, 2021. 1

[43] Guanghui Qin and Jason Eisner. Learning how to ask: Querying LMs with mixtures of soft prompts. In *Proc. NAACL*, 2021. 2

[44] 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 *Proc. ICML*, 2021. 1, 3, 4

[45] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *Technical Report*, 2019. 1

[46] Hanoona Rasheed, Muhammad Uzair Khattak, Muhammad Maaz, Salman Khan, and Fahad Shahbaz Khan. Fine-tuned clip models are efficient video learners. In *Proc. CVPR*, 2023. 2

[47] Manli Shu, Weili Nie, De-An Huang, Zhiding Yu, Tom Goldstein, Anima Anandkumar, and Chaowei Xiao. Test-time prompt tuning for zero-shot generalization in vision-language models. In *Proc. NeurIPS*, 2022. 4

[48] Chen Sun, Austin Myers, Carl Vondrick, Kevin Murphy, and Cordelia Schmid. VideoBERT: A joint model for video and language representation learning. In *Proc. CVPR*, 2019. 1

[49] Yi-Lin Sung, Jaemin Cho, and Mohit Bansal. Vl-adapter: Parameter-efficient transfer learning for vision-and-language tasks. In *Proc. CVPR*, 2022. 2

[50] Makarand Tapaswi, Yukun Zhu, Rainer Stiefelhagen, Antonio Torralba, Raquel Urtasun, and Sanja Fidler. MovieQA: Understanding stories in movies through question-answering. In *Proc. CVPR*, 2016. 1

[51] Alex Jinpeng Wang, Yixiao Ge, Rui Yan, Ge Yuying, Xudong Lin, Guanyu Cai, Jianping Wu, Ying Shan, Xiaohu Qie, and Mike Zheng Shou. All in one: Exploring unified video-language pre-training. In *Proc. CVPR*, 2023. 1

[52] Syed Talal Wasim, Muzammal Naseer, Salman Khan, Fahad Shahbaz Khan, and Mubarak Shah. Vita-CLIP: Video and text adaptive clip via multimodal prompting. In *Proc. CVPR*, 2023. 2

[53] Dejing Xu, Zhou Zhao, Jun Xiao, Fei Wu, Hanwang Zhang, Xiangnan He, and Yueting Zhuang. Video question answering via gradually refined attention over appearance and motion. In *Proc. ACM Multimedia*, 2017. 1, 3

[54] Jun Xu, Tao Mei, Ting Yao, and Yong Rui. MSR-VTT: A large video description dataset for bridging video and language. In *Proc. CVPR*, 2016. 1

[55] Antoine Yang, Antoine Miech, Josef Sivic, Ivan Laptev, and Cordelia Schmid. Just ask: Learning to answer questions from millions of narrated videos. In *Proc. ICCV*, 2021. 1, 3, 4

[56] Antoine Yang, Antoine Miech, Josef Sivic, Ivan Laptev, and Cordelia Schmid. Learning to answer visual questions from web videos. *IEEE TPAMI*, 2022. 1, 3, 4

[57] Antoine Yang, Antoine Miech, Josef Sivic, Ivan Laptev, and Cordelia Schmid. Zero-shot video question answering via frozen bidirectional language models. In *Proc. NeurIPS*, 2022. 1, 2, 3, 4

[58] Zhou Yu, Dejing Xu, Jun Yu, Ting Yu, Zhou Zhao, Yueting Zhuang, and Dacheng Tao. Activitynet-QA: A dataset for understanding complex web videos via question answering. In *Proc. AAAI*, 2019. 1, 3, 4

[59] 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 *Proc. CVPR*, 2022. 1, 4, 3

[60] Rowan Zellers, Ximing Lu, Jack Hessel, Youngjae Yu, Jae Sung Park, Jize Cao, Ali Farhadi, and Yejin Choi. MERLOT: Multimodal neural script knowledge models. In *Proc. NeurIPS*, 2021. 1, 2

[61] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Conditional prompt learning for vision-language models. In *Proc. CVPR*, 2022. 2

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

[63] Xizhou Zhu, Jinguo Zhu, Hao Li, Xiaoshi Wu, Hongsheng Li, Xiaohua Wang, and Jifeng Dai. Uni-perceiver: Pre-training unified architecture for generic perception for zero-shot and few-shot tasks. In *Proc. CVPR*, 2022. 2