

MiniGPT4-Video: Advancing Multimodal LLMs for Video Understanding with Interleaved Visual-Textual Tokens

Kirolos Ataallah¹ Xiaoqian Shen¹ Eslam Abdelrahman¹ Essam Sleiman²

Deyao Zhu¹ Jian Ding¹ Mohamed Elhoseiny¹

¹ KAUST ² Harvard University

Abstract

This paper introduces MiniGPT4-Video, a multimodal Large Language Model (LLM) designed specifically for video understanding. The model is capable of processing both temporal visual and textual data, making it adept at understanding the complexities of videos. Building upon the success of MiniGPT-v2, which excelled in translating visual features into the LLM space for single images and achieved impressive results on various image-text benchmarks, this paper extends the model’s capabilities to process a sequence of frames, enabling it to comprehend videos. MiniGPT4-video does not only consider visual content but also incorporates textual conversations, allowing the model to effectively answer queries involving both visual and text components. The proposed model outperforms existing state-of-the-art methods, registering gains of 4.22%, 1.13%, 20.82%, and 13.1% on the MSVD, MSRVTT, TGIF, and TVQA benchmarks respectively (see Figure 1). Our models and code have been made publicly available [here](#).

1. Introduction

In recent years, Large Language Models (LLMs) research has witnessed remarkable advancements, with prominent models like GPT-4 [1], Llama 2 [28], and Mistral [11] showcasing unprecedented capabilities in processing and generating textual data. However, typical LLMs are inherently limited to text-centric tasks and do not naturally capture the multimodal nature of human interaction with the world. While some strides have been made towards integrating images into LLMs, exemplified by works such as MiniGPT and LLaVa[7, 20, 33], the incorporation of temporal information from videos remains relatively underexplored and presents significant research challenges.

Unlike static images, videos present a temporal dimension, comprising sequences of frames, essential for understanding dynamic visual content alongside textual input. In this study, we endeavor to adapt LLMs to comprehend the

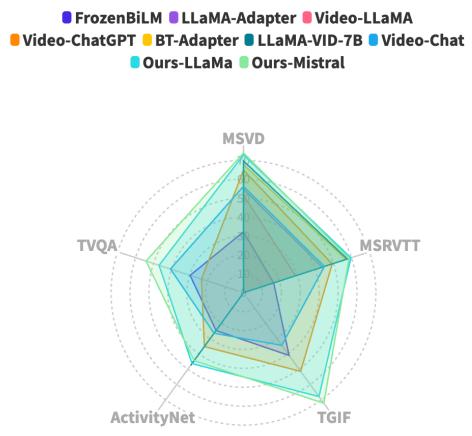


Figure 1. Holistic qualitative results across five video benchmarks, namely, MSVD, MSRVTTm TGIF, ActivityNet, and TVQA.

temporal intricacies inherent in video sequences. Previous efforts, such as Video-ChatGPT [22], have relied on spatial and temporal pooling techniques for fusing information across frames. However, these approaches often suffer from information loss and may not fully exploit the temporal dynamics of video data. Another example is LLaMA-VID [17], which attempted to address the constraints of LLMs in processing long videos by representing each frame with only two tokens, resulting in significant information loss.

In contrast, our approach leverages the concatenation of every four adjacent visual tokens, effectively reducing the token count while mitigating information loss. As depicted in Figure 2, we incorporate subtitles for each frame, allowing the representation of each frame as a combination of visual tokens extracted by the visual encoder[27] and text tokens derived from LLM tokenizers. Such an approach enables the LLM to comprehend the video content more comprehensively, facilitating responses to both visual and textual queries.

To validate the effectiveness of our proposed methodology, we conduct thorough evaluations across multiple benchmarks. These evaluations include assessments based on the

Video-ChatGPT benchmark [22], which evaluates aspects such as information correctness, detail orientation, contextual understanding, temporal comprehension, and consistency in video understanding. Additionally, we employ zero-shot evaluation methodologies encompassing open-ended questions and multiple-choice formats. As shown in Figure 1, the proposed MiniGPT4-Video outperforms existing state-of-the-art methods (7B) by notable margins of 4.22%, 1.13%, 20.82%, and 13.1% on the MSVD, MSRVT, TGIF, and TVQA benchmarks, respectively.

2. Related work

2.1. Large Vision-Language Models

The advancements in NLP and LLMs have inspired a wide range of methods and models known as vision language models (VLM) for understanding cross-modalities [8, 16, 34]. OpenAI’s CLIP [24] model aligns an image and language encoder on a large-scale dataset of image-text pairs with a contrastive loss, enabling it to perform multimodal-retrieval. More recent methods have harnessed the power of LLMs. For example, Flamingo [2] leverages training on web-scraped image-text pairs to reveal the in-context learning capabilities of LVLMs. Similarly, BLIP-2 [14], which integrates off-the-shelf frozen pre-trained image encoders and large language models to bridge the modality gap with a Querying Transformer. There is a trend to use large-scale instruction-tuning datasets to fine-tune LVLMs. For example, LLaVA [20] explores the instruction tuning of its vision language model on generated GPT-4 data, while MiniGPT-v2 [7] and InstructBLIP [19] use BLIP-2 [14] to construct their datasets. As LLMs continue to achieve better results in image understanding, recent work has begun scaling these models to the more challenging video domain.

2.2. LLM-Based Video Understanding

Recently, vision-language models such as LLaVA [20] have been extended to the video domain to process short videos 5 minutes on average or less, with similar capabilities such as visual question-answering and captioning. Video-LLaMA [31] and VideoChat [15] extend the BLIP-2 [14] architecture for video embedding extraction and both employ two streams for audio and visual signals. Video-LLaMA employs a Video Q-Former and an Audio Q-Former for the two streams, while VideoChat has a video embedder and a perception toolkit for captioning, tags, etc. On the other hand, Video-ChatGPT [22] leverages a single stream where the architecture first encodes each frame and then has a spatial and temporal pooling process that is finally mapped to an LLM with a linear layer. Video LLaVA [18] takes advantage of the LanguageBind module to map both image and video inputs to the same embedding space. Otter [13] proposed an instruction-tuned version of OpenFlamingo [3], such that it

can also process multiple video frames as input.

3. MiniGPT4-Video

3.1. Methodology

MiniGPT-v2 [7], has successfully translated visual features into the LLM space, enabling understanding of single images. However, extending this capability to multiple frames for video comprehension entails fine-tuning the LLM to process these frames and learn the temporal dynamics. Due to constraints imposed by the LLM’s context window, each video undergoes frame sub-sampling, with the number of frames (N) determined by the LLM’s context window. Subsequently, the visual frames are aligned with textual descriptions using a pre-trained model, EVA-CLIP [27], followed by a mapping into the large language model space using a linear layer. Similar to MiniGPT-v2 [7], we condense every four adjacent visual tokens in each image into a single token, thereby reducing token count per image by 75%, from 256 to 64. During training the subtitles are provided with the dataset but while inference or when there is no subtitle for the video, we utilize speech-to-text model such as whisper to generate the subtitles of the video. Frame subtitles are tokenized using the LLM tokenizer, and the visual and text tokens are concatenated for each sampled frame. Instruction tokens are appended to the end of the input sequence, and the model then outputs the answer to the question.

3.2. Training Pipeline

Large-scale image-text pair pretraining. In the first stage, we train a linear layer, similar as [33], which projects the visual feature encoded by the vision encoder (*e.g.* EVA-CLIP [27]) to the LLM’s text space with captioning loss. We leverage a combined image captioning dataset that includes images from LAION [25], Conceptual Captions [26], and SBU [23] to align the visual feature with LLM’s input space.

Large-scale video-text pair pretraining. In the second stage, we enable the model to understand videos by taking multiple frames as input. Specifically, we sample a maximum of N frames from each video. During this stage, we use the predefined prompts in the following template:
`<s>[INST]<FrameFeature_1><Sub><Subtitle text_1>... <FrameFeature_N><Sub><Subtitle text_N><Instruction></INST>`

The number of sampled frames is contingent upon the context window of each language model. Specifically, for Llama 2 [28], the context window is 4096 tokens, and Mistral [11] context window is 8192 tokens. In our approach, we represent each image with 64 tokens. Thus, for Llama 2, we designate N=45 frames, equating to 2880 tokens for visual content representation. Furthermore, we allocate 1000 tokens for subtitles, while the remaining tokens are earmarked for model output. Analogously, in the case

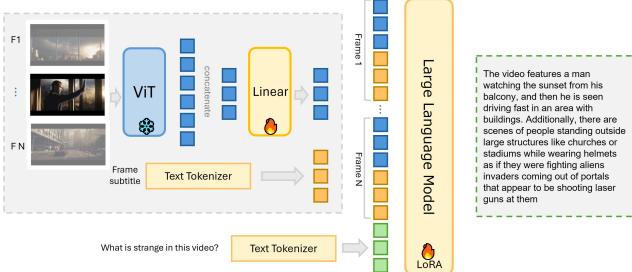


Figure 2. MiniGPT4-video architecture: For each frame, we use EVA-CLIP to get the visual tokens and concatenate each adjacent visual token into a singular token then convert these tokens to the language model space using a linear layer and get the language token from LLM tokenizer. Concatenate both the visual and subtitle text tokens together and do this for all the sampled frames and appending the instruction tokens at the end of the input sequence.

of Mistral, where the context window is doubled, N is doubled accordingly to $N=90$ frames, ensuring compatibility with the extended context window. In this prompt, each $\langle FrameFeature \rangle$ is replaced by the sampled video frame encoded by the visual backbone. The $\langle Subtitle\ text \rangle$ represents the subtitle for the corresponding frame if applicable, and $\langle Instruction \rangle$ represents a randomly sampled instruction from our predefined instruction set containing variant forms of instruction, such as “*Briefly describe this video*”. We use combined video captioning data incorporating CMD [4] and WebVid [5] for large-scale video captioning training.

Video question answering instruction finetuning. In this phase, we adopt the same training strategy implemented in the second stage but focus on leveraging high-quality video-question-answering datasets for instruction fine-tuning. This fine-tuning stage helps to enhance the model’s ability to interpret the input video and generate precise responses to the corresponding questions. The template is the same as the second stage with $\langle Instruction \rangle$ replaced by general questions as mentioned in the Video-ChatGPT [22] dataset.

3.3. Implementation Details

Throughout three training stages, we maintained a batch size of 4 and utilized the AdamW optimizer in conjunction with a cosine learning rate scheduler, setting the learning rate to 1e-4. Our visual backbone is EVA-CLIP [27], with the frozen weights. Notably, we trained the linear projection layer and performed efficient fine-tuning of the language model using LoRA [9]. Specifically, we fine-tuned the W_q and W_v components with a rank (r) of 64 and a LoRA-alpha value equal 16. The entire model was trained with a consistent image resolution of 224×224 pixels, ensuring uniformity across all stages.

Table 1. Quantitative results on Video-ChatGPT benchmark.

Method	Using Subtitles	Video ChatGPT				
		Information Correctness	Detailed Orientation	Contextual Understanding	Temporal Understanding	Consistency
LLaMA Adapter [32]	x	2.03	2.32	2.30	1.98	2.15
Video LLaMA [31]	x	1.96	2.18	2.16	1.82	1.79
Video Chat [15]	x	2.23	2.50	2.53	1.94	2.24
Video-ChatGPT [22]	x	2.40	2.52	2.62	1.98	2.37
BT-Adapter-7B [21]	x	2.68	2.69	3.27	2.34	2.46
LLaMA-VID-7B [17]	x	2.96	3.00	3.53	2.46	2.51
Ours Mistral-7B	x	2.97	2.58	3.17	2.38	2.44
Ours Llama 2-7B	x	2.93	2.97	3.45	2.47	2.60
Ours Mistral-7B	✓	3.05	2.61	3.24	2.42	2.42
Ours Llama 2-7B	✓	3.08	3.02	3.57	2.65	2.67

4. Experiments

4.1. Datasets

Training Datasets The Condensed Movies Video Captions dataset (CMD)[4] comprises approximately 15,938 videos, each spanning one to two minutes in length. However, CMD’s captions exhibit limited quality, characterized by an average sentence length of 14 words. The Webvid dataset[5] boasts a vast collection of two million videos. To align with CMD’s duration criteria, we refined this dataset to include videos ranging from one to two minutes in length. On the other hand, the Video Instruction Dataset [22] offers a rich resource of 100,000 question-answer pairs distributed across 13,224 videos, distinguished by meticulous annotations. Noteworthy for its high-quality annotations, this dataset presents detailed answers to questions, averaging 57 words per sentence. Spanning diverse question types, including Video Summarization and Description-based QAs, it addresses spatial, temporal, relationship, and reasoning aspects, alongside creative or generative QAs.

Evaluation Benchmarks The Video-ChatGPT benchmark [22], leveraging the ActivityNet-200 dataset [6], is meticulously designed to evaluate video-based conversation models’ text generation capabilities across five crucial dimensions: Correctness of Information, Detail Orientation, Contextual Understanding, Temporal Understanding, and Consistency. In assessing model performance on open-ended questions, established datasets such as MSRVTT-QA [29], MSVDQA [29], TGIF-QA FrameQA [10], and ActivityNet-QA [30] are employed. Furthermore, for multi-choice questions, model performance is scrutinized using the TVQA dataset [12], which is centered around popular TV shows. The validation set comprises 15,253 QA pairs, providing a robust framework for evaluation.

4.2. Evaluation Metrics

Aligned with the evaluation methodology established by Video-ChatGPT [22], we employed GPT-3.5 turbo to juxtapose model outputs with ground truth data, subsequently computing both accuracy and a score. The accuracy metric indicates the degree of correspondence between the model’s output and the ground truth, while the score ranges from 0 to 5, signifying the level of alignment between the model out-

Table 2. Zeroshot evaluation for open ended question and Multiple choices questions on MSVD, MSRVTT, TGIF, ActivityNet and TVQA.NA indicates not applicable, while MSVD and TGIF videos do not have audio.

Method	Using Subtitles	Open Ended Questions						MCQ	
		MSVD		MSRVTT		TGIF		ActivityNet	TVQA
		Acc. \uparrow	Score \uparrow	Acc. \uparrow	Score \uparrow	Acc. \uparrow	Score \uparrow	Acc. \uparrow	Score \uparrow
FrozenBiLM	X	32.2	—	16.8	—	41	—	24.7	—
LLaMA Adapter [32]	X	54.9	3.1	43.8	2.7	—	—	34.2	2.7
Video LLaMA [31]	X	51.6	2.5	29	1.8	—	—	12.4	1.1
Video Chat [15]	X	56.3	2.8	45	2.5	34.4	2.3	26.5	2.2
Video-ChatGPT [22]	X	64.9	3.3	49.3	2.8	51.4	3.0	35.2	2.7
BT-Adapter-7B [21]	X	67.7	3.7	57	3.2	—	—	45.7	3.2
LLaMA-VID-7B [17]	X	69.7	3.7	57.7	3.2	—	—	47.4	3.3
Ours Llama 2-7B	X	72.93	3.84	58.83	3.29	67.9	3.71	45.85	3.23
Ours Mistral-7B	X	73.92	4.06	58.26	3.52	72.22	4.08	44.25	3.35
Ours Llama 2-7B	✓	N/A	N/A	59.73	3.3	N/A	N/A	46.3	3.4
Ours Mistral-7B	✓	N/A	N/A	58.68	3.53	N/A	N/A	44.38	3.36
									54.21

put and the ground truth. A score of 0 indicates a significant deviation from the ground truth, while a score of 5 suggests close alignment. To ensure a fair and consistent comparison with the results presented in Video-ChatGPT [22], we adopted the same prompt for our evaluations.

4.3. Results

For a comprehensive evaluation of our proposed architecture, we assessed its performance across three benchmark types: Video-ChatGPT, Open-ended Questions, and Multiple-Choice Questions (MCQs). In the Video-ChatGPT benchmark, depicted in Table 1, our model is comparable with the previous methods without subtitles. When we add the subtitles as input, our model achieves the state-of-the-art in all five dimensions, which verified that our model can utilize the subtitle information to improve the video understanding. In the zero-shot evaluation of open-ended and multiple-choice question benchmarks, as illustrated in Figure 1 and Table 2, our proposed MiniGPT4-Video significantly outperforms existing state-of-the-art methods. It achieves notable margins of improvement 4.22%, 1.13%, 20.82%, and 13.1% on the MSVD, MSRVTT, TGIF, and TVQA benchmarks, respectively. The results, both with and without subtitles as shown in Table 2, further demonstrate that integrating subtitle information alongside visual cues significantly enhances performance, with accuracy rising from 33.9% to 54.21% on TVQA. While subtitles contribute substantially to performance improvements on TVQA, their inclusion doesn't offer added value for datasets like MSVD-QA, MSRVTT-QA, TGIF-QA, and ActivityNet, where questions are exclusively vision-based.

5. Qualitative Results

Here in this section we show some qualitative results for our model to show the performance of it and its ability to answer different questions. For each example you can open the video link which attached to the figure description to watch the video.

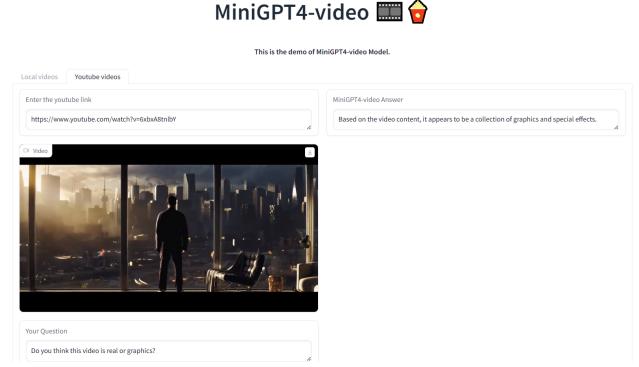


Figure 3. Qualitative result of MiniGPT4-video for understanding general videos [link].

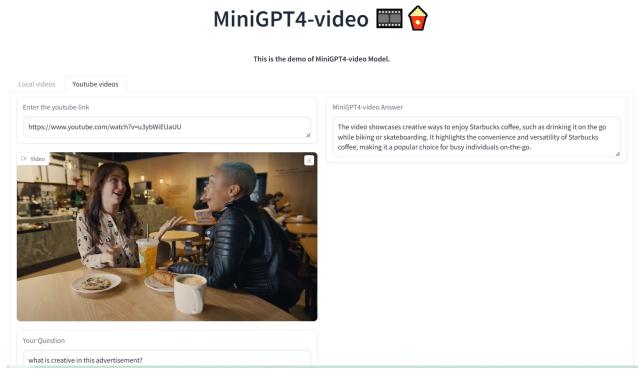


Figure 4. Qualitative result of MiniGPT4-video [link].

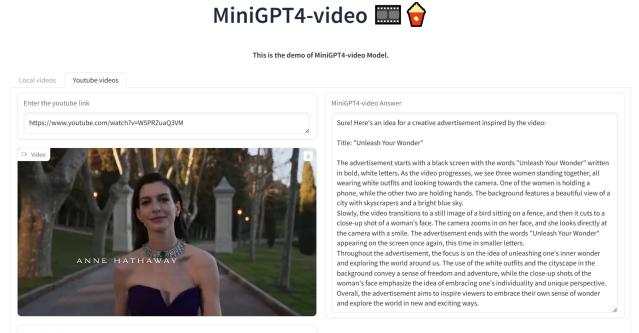


Figure 5. Qualitative result of MiniGPT4-video [link].

6. Conclusion

In summary, MiniGPT4-Video offers a compelling solution for video question answering, effectively amalgamating visual and conversational comprehension within the video domain. By directly inputting both visual and textual tokens, MiniGPT4-Video empowers the Language Modeling Model (LLM) to grasp the intricate relationships between video frames, showcasing promising proficiency in understanding temporal dynamics within video content. Despite its notable achievements, MiniGPT4-Video faces a limitation

MiniGPT4-video 📽️💡

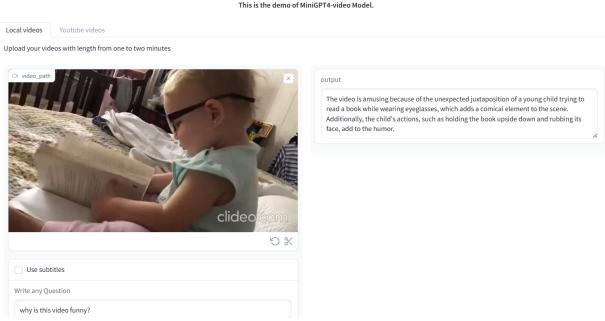


Figure 6. Qualitative result of MiniGPT4-video

MiniGPT4-video 📽️💡

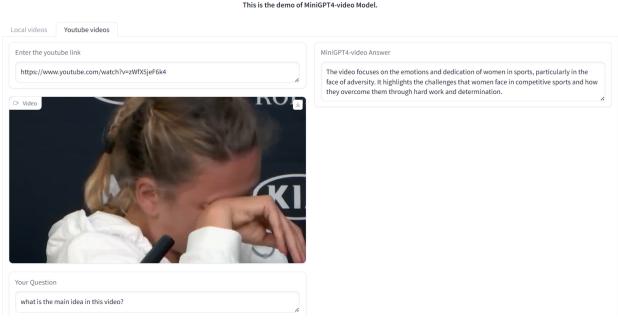


Figure 9. Qualitative result of MiniGPT4-video [link].

MiniGPT4-video 📽️💡

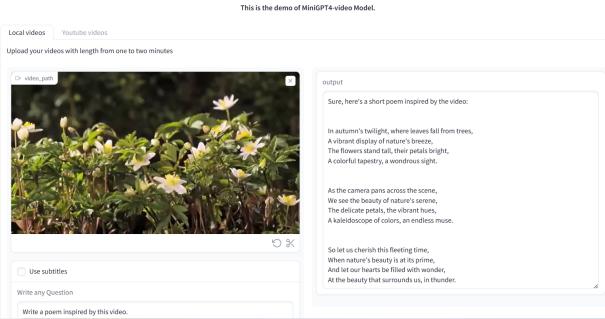


Figure 7. Qualitative result of MiniGPT4-video



Figure 8. MiniGPT4-video vs VideoChatGPT [22].

imposed by the context window of the LLM. Specifically, the current version requires video lengths of 45 frames for the Llama 2 version (equivalent to less than one and a half minutes at a sampling rate of 0.5 frames per second) and 90 frames for the Mistral version (equivalent to less than three minutes). Future research endeavors will focus on extending the model's capabilities to handle longer video sequences, thereby addressing this limitation and further enhancing its applicability and effectiveness in real-world scenarios.

References

- [1] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023. [1](#)
- [2] 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. *Advances in Neural Information Processing Systems*, 35:23716–23736, 2022. [2](#)
- [3] Anas Awadalla, Irena Gao, Josh Gardner, Jack Hessel, Yusuf Hanafy, Wanrong Zhu, Kalyani Marathe, Yonatan Bitton, Samir Gadre, Shiori Sagawa, et al. Openflamingo: An open-source framework for training large autoregressive vision-language models. *arXiv preprint arXiv:2308.01390*, 2023. [2](#)
- [4] Max Bain, Arsha Nagrani, Andrew Brown, and Andrew Zisserman. Condensed movies: Story based retrieval with contextual embeddings. In *Proceedings of the Asian Conference on Computer Vision*, 2020. [3](#)
- [5] Max Bain, Arsha Nagrani, Gülcin Varol, and Andrew Zisserman. Frozen in time: A joint video and image encoder for end-to-end retrieval. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1728–1738, 2021. [3](#)
- [6] Fabian Caba Heilbron, Victor Escorcia, Bernard Ghanem, and Juan Carlos Niebles. Activitynet: A large-scale video benchmark for human activity understanding. In *Proceedings of the ieee conference on computer vision and pattern recognition*, pages 961–970, 2015. [3](#)
- [7] Jun Chen, Deyao Zhu, Xiaoqian Shen, Xiang Li, Zechun Liu, Pengchuan Zhang, Raghuraman Krishnamoorthi, Vikas Chandra, Yunyang Xiong, and Mohamed Elhoseiny. Minigpt-v2: large language model as a unified interface for vision-language multi-task learning. *arXiv preprint arXiv:2310.09478*, 2023. [1, 2](#)
- [8] Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. Uniter: Universal image-text representation learning. In *European conference on computer vision*, pages 104–120. Springer, 2020. [2](#)

- [9] 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. 3
- [10] Yunseok Jang, Yale Song, Youngjae Yu, Youngjin Kim, and Gunhee Kim. Tgif-qa: Toward spatio-temporal reasoning in visual question answering, 2017. 3
- [11] Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. Mistral 7b. *arXiv preprint arXiv:2310.06825*, 2023. 1, 2
- [12] Jie Lei, Licheng Yu, Mohit Bansal, and Tamara L. Berg. Tvqa: Localized, compositional video question answering, 2019. 3
- [13] Bo Li, Yuanhan Zhang, Liangyu Chen, Jinghao Wang, Jingkang Yang, and Ziwei Liu. Otter: A multi-modal model with in-context instruction tuning. *arXiv preprint arXiv:2305.03726*, 2023. 2
- [14] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. *arXiv preprint arXiv:2301.12597*, 2023. 2
- [15] KunChang Li, Yinan He, Yi Wang, Yizhuo Li, Wenhui Wang, Ping Luo, Yali Wang, Limin Wang, and Yu Qiao. Videochat: Chat-centric video understanding, 2024. 2, 3, 4
- [16] 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
- [17] Yanwei Li, Chengyao Wang, and Jiaya Jia. Llama-vid: An image is worth 2 tokens in large language models, 2023. 1, 3, 4
- [18] Bin Lin, Bin Zhu, Yang Ye, Munan Ning, Peng Jin, and Li Yuan. Video-llava: Learning united visual representation by alignment before projection. *arXiv preprint arXiv:2311.10122*, 2023. 2
- [19] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. *arXiv preprint arXiv:2310.03744*, 2023. 2
- [20] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *arXiv preprint arXiv:2304.08485*, 2023. 1, 2
- [21] Ruyang Liu, Chen Li, Yixiao Ge, Ying Shan, Thomas H. Li, and Ge Li. One for all: Video conversation is feasible without video instruction tuning, 2023. 3, 4
- [22] Muhammad Maaz, Hanoona Rasheed, Salman Khan, and Fahad Shahbaz Khan. Video-chatgpt: Towards detailed video understanding via large vision and language models, 2023. 1, 2, 3, 4, 5
- [23] Vicente Ordonez, Girish Kulkarni, and Tamara Berg. Im2text: Describing images using 1 million captioned photographs. *Advances in neural information processing systems*, 24, 2011. 2
- [24] 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 *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 2
- [25] Christoph Schuhmann, Richard Vencu, Romain Beaumont, Robert Kaczmarek, Clayton Mullis, Aarush Katta, Theo Coombes, Jenia Jitsev, and Aran Komatsuzaki. Laion-400m: Open dataset of clip-filtered 400 million image-text pairs, 2021. 2
- [26] Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2556–2565, 2018. 2
- [27] Quan Sun, Yuxin Fang, Ledell Wu, Xinlong Wang, and Yue Cao. Eva-clip: Improved training techniques for clip at scale. *arXiv preprint arXiv:2303.15389*, 2023. 1, 2, 3
- [28] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajwala Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023. 1, 2
- [29] Dejing Xu, Zhou Zhao, Jun Xiao, Fei Wu, Hanwang Zhang, Xiangnan He, and Yuetong Zhuang. Video question answering via gradually refined attention over appearance and motion. In *Proceedings of the 25th ACM international conference on Multimedia*, pages 1645–1653, 2017. 3
- [30] Zhou Yu, Dejing Xu, Jun Yu, Ting Yu, Zhou Zhao, Yuetong Zhuang, and Dacheng Tao. Activitynet-qa: A dataset for understanding complex web videos via question answering, 2019. 3
- [31] Hang Zhang, Xin Li, and Lidong Bing. Video-llama: An instruction-tuned audio-visual language model for video understanding. *arXiv preprint arXiv:2306.02858*, 2023. 2, 3, 4
- [32] Renrui Zhang, Jiaming Han, Aojun Zhou, Xiangfei Hu, Shilin Yan, Pan Lu, Hongsheng Li, Peng Gao, and Yu Qiao. Llama-adapter: Efficient fine-tuning of language models with zero-init attention. *arXiv preprint arXiv:2303.16199*, 2023. 3, 4
- [33] Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*, 2023. 1, 2
- [34] Mingchen Zhuge, Dehong Gao, Deng-Ping Fan, Linbo Jin, Ben Chen, Haoming Zhou, Minghui Qiu, and Ling Shao. Kaleido-bert: Vision-language pre-training on fashion domain. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12647–12657, 2021. 2