Language Models with Image Descriptors are Strong Few-Shot Video-Language Learners

Anonymous Author(s)
Affiliation
Address
email

Abstract

The goal of this work is to build flexible video-language models that can generalize to various video-to-text tasks from few examples, such as domain-specific captioning, question answering, and future event prediction. Existing few-shot video-language learners focus exclusively on the encoder, resulting in the absence of a video-to-text decoder to handle generative tasks. Video captioners have been pretrained on large-scale video-language datasets, but they rely heavily on finetuning and lack the ability to generate text for unseen tasks in a few-shot setting. We propose **VidIL**, a few-shot **Vid**eo-language Learner via **I**mage and **L**anguage models, which demonstrates strong performance on few-shot video-to-text tasks without the necessity of pretraining or finetuning on any video datasets. We use the image-language models to translate the video content into frame captions, object, attribute, and event phrases, and compose them into a temporal structure template. We then instruct a language model, with a prompt containing a few in-context examples, to generate a target output from the composed content. The flexibility of prompting allows the model to capture any form of text input, such as automatic speech recognition (ASR) transcripts. Our experiments demonstrate the power of language models in understanding videos on a wide variety of video-language tasks, including video captioning, video question answering, video caption retrieval, and video future event prediction. Especially, on video future event prediction, our few-shot model significantly outperforms state-of-the-art supervised models trained on large-scale video datasets.

1 Introduction

One major gap between artificial intelligence and human intelligence lies in their abilities to generalize and perform well on new tasks with limited annotations. Recent advances in large-scale pre-trained generative language models [43, 6, 69, 24] have shown promising few-shot capabilities [70, 41, 61] in understanding natural language. However, few-shot video-language understanding is still in its infancy. A particular limitation of most recent video-language pretraining frameworks [28, 21, 59, 66, 65, 25, 62] is that they are encoder-only, which means they do not have the ability to generate text from videos for purposes such as captioning [60, 55], question answering [58], and future prediction [23]. Meanwhile, unified video-language models [34, 47] that are capable of language decoding still rely heavily on finetuning using a large number of manually annotated video-text pairs, therefore cannot adapt quickly to unseen tasks. Few-shot video-to-text decoding is challenging because the natural language supervision for learning video-language representation is typically based on subtitles and automatic speech recognition (ASR) transcripts [37, 66], which differ significantly from downstream tasks in terms of distribution and may have poor semantic alignment across vision and text modalities.

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

We propose to address this problem by harnessing the few-shot power of large-scale language models, such as GPT-3 [6] and InstructGPT [38]. Our inspiration is derived from the fact that humans are excellent visual storytellers [15], with the ability to piece together a coherent story from a few isolated images. To mimic this, we propose **VidIL**, a few-shot **Vid**eo-language Learner via **I**mage and **L**anguage models, to use image models to provide information about the visual content in the video (as well as optionally use ASR to represent speech), and then we instruct language models to interpret a video-based summary, answer, or other target output for diverse video-language tasks.

Videos contain rich semantics and temporal content at multiple granularities. Similar to static images, videos depict objects, attributes, and events. However, the sequence of frames further conveys object state changes, actions, and events. For example, in Figure 1, the frame captions describe static visual features such as *"a person holding a green object in hand"* for the first frame. In contrast, the video clip could be correctly captioned as *"a woman makes realistic looking leaves and flowers for a cake"*, or represented as a collection of the objects and events that occur at different timestamps in the video clip, such as *cutting mat* and *flowered design*. Hence, to inform video-level description and queries, we need to represent all of this information and its temporal ordering.

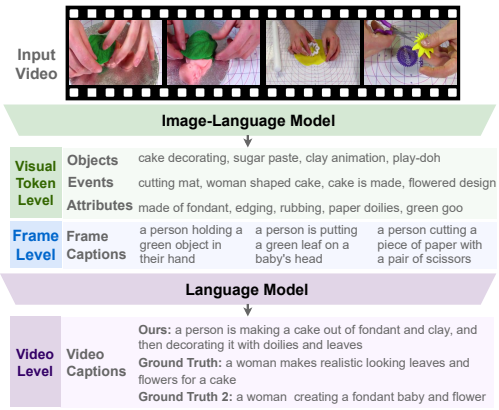

Figure 1: Multiple levels of information in videos.

To address these requirements, we propose to decompose a video into three levels: the video output, frame captions, and visual tokens (including object, event, attribute). One major benefit from this hierarchical video representation is that we can separate the visual and temporal dimensions of a video. We leverage well-trained image-language foundational models at lower levels to collect salient visual features from the sparsely sampled frames. Specifically, we leverage pretrained image-language contrastive model CLIP [42] to perform visual tokenization, based on the similarity score between frames and tokens of objects, events and attributes. The tokenization is done under the guidance of semantics role labeling [14], which provides us with candidate events with involved objects and related attributes. Next, in order to capture the overall semantics at the frame level, we employ the pretrained image captioner in image-language model BLIP [26] to obtain frame captions. We then instruct pretrained language models using in-context learning [38, 13, 49, 46] to interpret visual tokens and frame captions into the target text content. In detail, we temporally order visual tokens and frame captions using specially designed prompts such as *"First...Then...Finally"*, to instruct the pretrained language model to track the changes of objects, events, attributes and frame semantics along the temporal dimension.

We show that our approach outperforms both video-language and image-language state-of-the-art baselines on few-shot video captioning and question answering tasks. Moreover, on the video-language event prediction task, we significantly outperform fully-supervised models while using only 10 labeled examples. We further demonstrate that our generative model can benefit broader video-language understanding tasks, such as text-video retrieval, via pseudo label generation. Additionally, we show that our model is highly flexible in adding new modalities, such as ASR transcripts.

2 Related Work

2.1 Image-Language Models and Their Applications on Video-Language Tasks

Large-scale image-language pretraining models optimize image-text matching through contrastive learning such as CLIP [42] or by learning image-text alignments [63, 27, 56, 64, 33, 50, 8, 29, 71, 68, 17, 18, 16]. Recently, BLIP [26] proposes an image captioner pretrained with filtering in order to achieve optimal image-text alignment and has shown promising performance on various image-language tasks. However, video-language pretraining [25, 34, 28, 36, 3, 1, 40] is still hindered by noisy and domain-specific video datasets [22, 37]. Naturally, researchers start to explore transferring

the rich knowledge from image models to videos. Different from the traditional way of representing videos by 3D dense features [12], recent work [21, 25] proves that sparse sampling is an effective way to represent videos, which facilitates applying pre-trained image-language models to video-language tasks [35, 11]. Specifically, the image-language model BLIP [26] sets new state-of-the-art on zero-shot retrieval-style video-language tasks, such as video retrieval and video question answering. However, for generation-style tasks such as domain-specific video captioning, video-language model such as UniVL [34] still leads the performance but highly rely on fine-tuning. In this work, we extend the idea of leveraging image-language models to a wide variety of video-to-text generation tasks. We further connect image-language models (ILM) with language models (LM) which empowers our model with strong generalization ability. We show that the knowledge from both image-language pretraining and language-only pretraining can benefit video-language understanding in various aspects.

2.2 Unifying Multi-Modal Tasks with Language Models

Connecting different modalities with an unified representation has been paid much attention to recently. Text-only generation models, such as T5 [44], has been extended to vision-language tasks by text generation conditioned on visual features [9, 51, 48, 73, 53]. In order to fully leverage the generalization power from pretained language models, [61] represents images using text in a fully symbolic way. [32] includes more modalities such as video and audio, but requires annotated video-text data to jointly training the language model with the video and audio tokenizer. In this work, we propose a temporal-aware hierarchical representation for describing a video textually. To our knowledge, we are the first work to leverage prompting a frozen language model for tackling few-shot video-language tasks with a unified textual representation. Concurrent work Socratic [67] uses a zero-shot language-based world-state history to represent long videos with given time stamps, while our model can quickly adapt to different video and text distributions with few examples. Furthermore, we show that by injecting *temporal markers* to the prompt we can make a pre-trained language model understand fine-grained temporal dynamics in video events. Compared with the concurrent work Flamingo [2], which requires dedicated vision-language post-pretraining, our framework does not require to pretrain on any videos. Our framework is simple and highly modulated where all the components are publicly available. Additionally, our framework is more flexible on adding new modalities, e.g., automatic speech recognition, without the need for complex redesigning.

3 Method

We propose a hierarchical video representation framework which decomposes a video into three levels, i.e., **visual token level**, **frame level** and **video level**. The motivation is to separate the spatial and temporal dimension of a video in order to leverage image-language and language-only foundation models, such as CLIP [42] and GPT-3 [6]. All three levels use a unified textual representation which enables us to leverage the powerful few-shot ability from pretrained language models.

3.1 Frame Level: Image Captioning

Following [21] we first perform sparse sampling to obtain several video frames. Unless otherwise specified, we sample 4 frames for frame level and 8 frames for visual token level. We then feed each frame into a pre-trained image-language model to obtain frame level captions. An example can be found in the blue part of Figure 2. In our experiments, we use BLIP [26], a recent image-language framework containing both image-grounded encoder and decoder, for generating frame captions. We follow [26] to do both captioning and filtering on each frame. However, as mentioned in Section 1, videos contain rich semantics and temporal contents at multiple granularities. It is not enough to generate video-level target text such as video captions solely based on frame captions. Thus, we further perform visual tokenization for each frame to capture features at a finer granularity.

3.2 Visual Token Level: Structure-Aware Visual Tokenization

At this level, we aim to extract the textual representations of salient visual token types, such as objects, events and attributes. We found that pre-defined classes for classification, such as those in ImageNet [10], are far from enough for covering the rich semantics in open-domain videos. Thus, instead of using classification-based methods for visual tokenization as in previous work [32, 61],

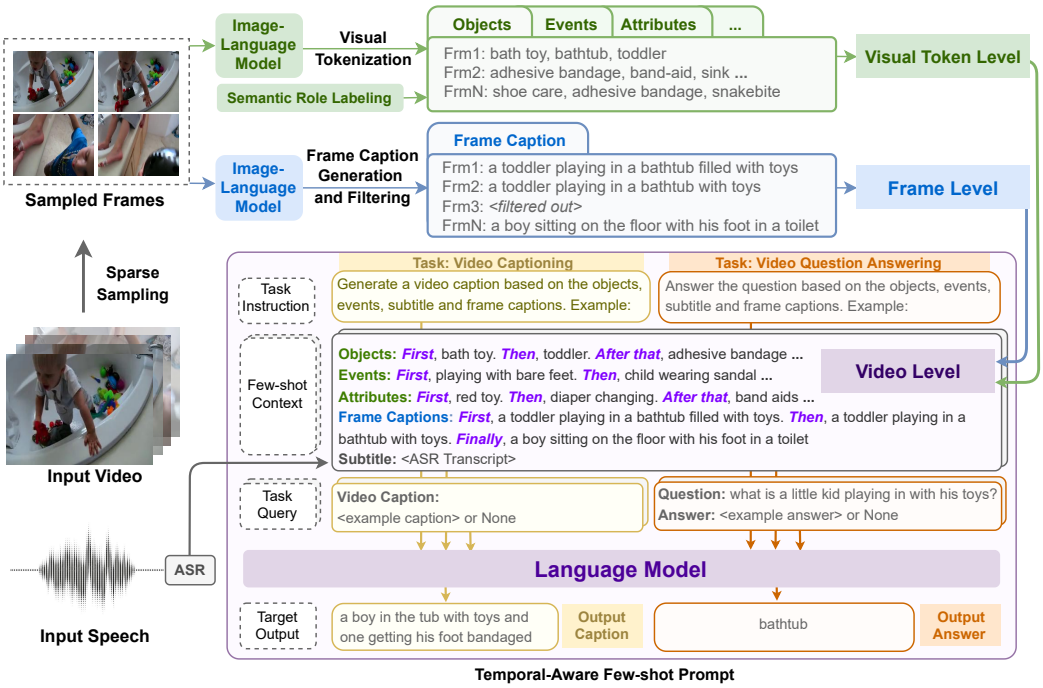

Figure 2: Overview of VidIL framework. We represent a video in a unified textural representation containing three semantic levels: **visual token level**, **frame level**, and **video level**. At visual token level, we extract salient objects, events, attributes for each sampled frame. At frame level, we perform image captioning and filtering. And at video level, we construct the video representation by aggregating the visual tokens, frame captions and other text modalities such as **ASR**, using a few-shot temporal-aware prompt. We then feed the prompt to a pre-trained language model together with task-specific instructions to generate target text for a variety of video-language tasks.

we adopt a retrieval-based visual tokenization approach by leveraging pre-trained contrastive image-language models. Given a visual token vocabulary which contains all candidate object, event, and attribute text phrases, we compute the image embedding of a frame and the text embeddings of the candidate visual tokens using a contrastive multi-modal encoder, CLIP [42]. We then select top 5 visual tokens based on the cosine similarity of the image and text embeddings. An example of the extracted object tokens can be found in the green part of Figure 2.

Unlike in images where objects and attributes already cover most visual features, events are more informative in videos. In order to discover events from video frames, we construct our own event vocabulary by extracting event structures from Visual Genome [19] synsets[1] using **Semantic Role Labeling**. Specifically, we first select the phrases that contains at least one verb and one argument as **events**. Then we remove highly similar events based on their sentence similarity using Sentence-BERT [45] embeddings. For object vocabulary, we adopt OpenImage [20] full classes (∼20k), instead of using the visually groundable subset (∼600) as in concurrent work [67]. We found that using *large but noisy* vocabulary is more effective than using *small but clean* vocabulary in our retrieval-based setting with CLIP. For attribute vocabulary, we adopt visual genome attribute synset. In Section 4.6, we provide ablation study on the impact of different types of visual tokens. The statistics of visual token vocabulary can be found in Appendix.

3.3 Video Level: Temporal-Aware Few-shot Prompting

Once we obtain the textual representation from frame level and visual token level, the final step is to put the pieces together to generate a video level target text. The goal is to build a model that can be quickly adapted to any video-to-text generation task with only a few examples. To this end, we propose to leverage large-scale pre-trained language models, such as GPT-3 [6], with a

[1] We use the keys in Visual Genome [19] object synsets which contains frequent <verb,object> pairs.

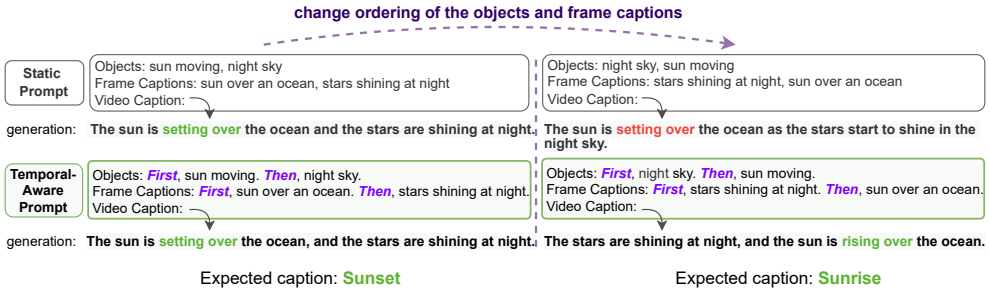

Figure 3: Temporal-aware prompt successfully distinguishes the **Sunset** and **Sunrise** scenario from the temporal ordering change of objects and frame captions, while the static prompt fails.

temporal-aware few-shot prompt. As shown in Figure 2, our framework can be readily applied to various video-to-text generation tasks, such as video captioning and video question answering, with a shared prompt template. The proposed prompting strategy enables a language model to attend to the lower level visual information as well as taking into account the temporal ordering.

Here, we use the video captioning task depicted in Figure 2 to illustrate the details. The few-shot prompt consists of three parts: **instruction**, **few-shot context**, and **task query**. The **instruction** is a concise description of the generation task, e.g., `"Generate a video caption based on the objects, events, attributes and frame captions. Example:"`, which is proved to be effective in zero-shot and few-shot settings [6, 57]. The **few-shot context** contains the selected in-context examples as well as the test video instance. Each video instance is represented by the aggregated visual tokens[2], e.g., `"Objects: First, bath toy. Then,..."`, the frame captions, e.g., `"Frame Captions: First, a toddler playing in a bathtub filled with toys. Then,..."`, and the ASR inputs if available, e.g., `"Subtitle:<ASR Transcript>"`. Finally, the **task query** is a task-specific suffix indicating the target text format, e.g. `"Video Caption:"`. For in-context examples (omitted here for simplicity), the task query is followed by ground truth annotation, while for the test instance, the generation starts at the end of the task query.

Formally, we denote the instruction line as $\mathbf{t}$, few-shot context as $\mathbf{c}$, the task query as $\mathbf{q}$, and the target text as $\mathbf{y}$, where $\mathbf{y} = (y_1, y_2, ..., y_L)$. The generation of the next target token y_l can be modeled as:

$$y_l = \arg\max_{y} p(y|\mathbf{s}, \mathbf{c}, \mathbf{q}, y_{<l}) \tag{1}$$

In order to capture the temporal dynamics between frames and visual tokens, we further propose to inject **temporal markers** to the prompt. As shown in the few-shot context in Figure 2, each visual token and frame caption is prefixed with a natural language phrase indicating its temporal ordering, e.g., `"First,"`,`"Then,"`, and `"Finally,"`. We found adding the temporal marker can make the language model condition on not only the literal but also the temporal information of the context. We show an example in Figure 3, where we compare our *temporal-aware prompt* with a *static prompt* on video captioning using InstructGPT. Again, the in-context examples are omitted here, which can be found in Appendix. In this example, the only difference between the two context is the ordering of the visual tokens and the frame captions. For the context on the left, where `"sun moving"` appears before `"night sky"`, we are expected to see a story talking about **sunset**, while for the context on the right, we are expected to see **sunrise**. We can see the static prompt generates captions about sunset for both contexts, while the temporal-aware prompt can capture the temporal ordering correctly and generate sunrise for the context on the right.

4 Experiments

4.1 Experimental Setup

To comprehensively evaluate our model, we show results on four video-language understanding tasks in few-shot settings: video captioning, video question answering (QA), video-language event

[2]To obtain video level visual tokens, the visual tokens extracted from each frame are further ranked and ordered based on frequency and frame index.

prediction, and text-video retrieval. We compare our approach with state-of-the-art approaches on five benchmarks, i.e, MSR-VTT [60], MSVD [7], VaTeX [55], YouCook2 [72], and VLEP [23].

Implementation Details. We use CLIP-L/14[4] as our default encoder for visual tokenization. We adopt BLIP captioning checkpoint[5] fine-tuned on COCO [31] for frame captioning. We use InstructGPT [38] as our default language model for generating text conditioned on the few-shot prompt. To construct event vocabulary, we use the semantic role labeling model from AllenNLP[6]. The experiments are conducted on 2 NVIDIA V100 (16GB) GPUs.

Table 1: Statistics of datasets in our experiments

Dataset	Task	Split Count # train / # eval
MSR-VTT [60]	Captioning; QA	6,513 / 2,990
MSR-VTT [60]	Retrieval	7,010 / 1,000
MSVD [7]	Question Answering	30,933 / 13,157
VaTeX v1.1[3] [55]	Captioning; Retrieval	25,991 / 6,000
YouCook2 [72]	Captioning	10,337 / 3,492
VLEP [23]	Event Prediction	20,142 / 4,192

In-context Example Selection. From our preliminary experiments, we found that the generation performance is sensitive to the quality of in-context examples. For example, for QA tasks such as MSVD-QA where the annotations are automatically generated, the <question, answer> pair in randomly selected in-context examples can be only weakly-correlated with the video context. Thus, instead of using a fixed prompt for each query, we dynamically filter out the irrelevant in-context examples. Specifically, given a randomly sampled *M*-shot support set from the training set, we select a subset of *N*-shots as in-context examples based on their SentenceBERT [45] similarities with text queries. For QA tasks, we choose the most relevant in-context examples by comparing with questions. While for captioning task, we compare with frame captions. If not otherwise specified, we use *M=10* and *N=5*, which we consider as 10-shot training.

4.2 Few-shot Video Captioning

We report BLEU-4 [39], ROUGE-L [30], METEOR [5], and CIDEr [52] scores on three video captioning benchmarks covering both open-domain (MSR-VTT, VaTeX) and domain-specific (YouCook2) videos. We compare with both state-of-the-art video captioner (UniVL [34]) and image captioner (BLIP [26]). In order to implement the BLIP baseline for few-shot video captioning, we extend the approach used for text-video retrieval evaluation in [26] to video-language training. Specifically, we concatenate the visual features of sampled frames and then feed them into the image-grounded text-encoder to compute the language modeling loss. This is equivalent to stitching the sampled frames into a large image and then feeding it to BLIP for image captioning. We found that this simple approach results in very strong baselines.

Table 2: 10-shot video captioning results. *B-4*, *R-L*, *M*, *C* represents *BLEU-4*, *ROUGE-L*, *METEOR* and *CIDEr*. *Avg C* represents the average CIDEr score across all available benchmarks. *ASR* indicates whether the model has access to the ASR subtitles. *BLIP* and $BLIP_{cap}$ use the pretrained checkpoint and the finetuned checkpoint on COCO captioning. All results are averaged over three random seeds.

Method	ASR	MSR-VTT Caption				YouCook2 Caption				VaTex Caption				Avg C
		B-4	R-L	M	C	B-4	R-L	M	C	B-4	R-L	M	C	
Few-shot														
UniVL	No	2.1	22.5	9.5	3.6	**3.3**	**25.3**	**11.6**	**34.1**	1.7	15.7	8.0	2.1	13.3
BLIP	No	**27.7**	43.0	23.0	**39.5**	0.7	9.0	3.4	11.5	13.5	39.5	15.4	20.7	23.9
$BLIP_{cap}$	No	21.6	48.0	22.7	30.2	3.7	8.6	3.8	9.4	20.7	41.5	17.4	28.9	22.8
VidIL(ours)	No	26.0	**51.7**	**24.7**	36.3	2.6	22.9	9.5	27.0	**22.2**	**43.6**	**20.0**	**36.7**	**33.3**
UniVL	Yes	-	-	-	-	4.3	26.4	12.2	48.6	2.7	17.7	10.2	3.4	26.0
VidIL(ours)	Yes	-	-	-	-	**10.7**	**35.9**	**19.4**	**111.6**	**23.2**	**44.2**	**20.6**	**38.9**	**75.3**
Fine-tuning														
UniVL	No	42.0	61.0	29.0	50.1	11.2	40.1	17.6	127.0	22.8	38.6	22.3	33.4	70.2
UniVL	Yes	-	-	-	-	16.6	45.7	21.6	176.8	23.7	39.3	22.7	35.6	106.2

[4] https://huggingface.co/openai/clip-vit-large-patch14

[5] https://github.com/salesforce/BLIP#finetuned-checkpoints

[6] https://docs.allennlp.org/models/main/models/structured_prediction/predictors/srl/

As shown in Table 2, existing methods have strong bias on certain datasets. For example, UniVL performs well on YouCook2 but fails on MSR-VTT and VaTeX, while BLIP performs the opposite. This is because UniVL is pretrained on HowTo100M which favors instructional videos, i.e., YouCook2, while BLIP is pre-training on image-caption pairs which favors description-style captions, i.e., MSR-VTT and VaTeX. On the contrary, our model performs competitively on both open-domain and instructional videos, and significantly outperforms the baselines in the average CIDEr score across all three benchmarks. This indicates that by leveraging language models, we can maintain strong few-shot ability regardless of the video domain and or the target caption distribution.

As discussed in Section 1, video captions describes the content in various semantic levels. The N-gram based metric may not fairly reflect the models' performance in capturing the video-caption alignment. We further verifies this hypothesis in Section 4.5. Thus, in addition to the automatic metrics, we include qualitative examples illustrated in Figure 4. More examples are in Appendix.

Additionally, for most existing methods and also concurrent work, e.g., Flamingo [2], adding a new modality often requires a dedicated model redesign and or retraining. However, the nature of our framework, where we use a unified textual representation for each level, making it highly flexible for incorporating new modalities. As shown in row 5 and 6 in Table, our model can benefit from the added modality more effectively than the baselines.

Figure 4: Qualitative examples on video captioning. Grey boxes contains part of the video representation from our model. Blue boxes contains caption generation from different models. Green boxes contains ground truth annotations. **Bold green text** highlights the correct information that is not captured in baseline outputs which can be reasoned from our visual tokens and frame captions.

4.3 Few-shot Video Question Answering

We compare the test accuracy of our approach with few-shot pretrained BLIP, BLIP$_{VQA}$ [26], and concurrent work Flamingo [2] on two video question answering benchmarks, MSR-VTT_QA and MSVD_QA. BLIP$_{VQA}$ represents finetuned BLIP on VQA [4] dataset, which is the previous SOTA on zero/few-shot video question answering. In order to have fairer comparison with BLIP$_{VQA}$, we reduce the shot number to 5 and report the average accuracy on three sets of randomly selected 5-shot examples. As shown in Table 3, our method outperforms previous SOTA by a large margin. Comparing with concurrent work Flamingo, which is post-pretrained on a large number of video-text data, our model is training-free and did not observe any video data. However, with only image-language and language-only knowledge, our 5-shot model is able to beat 8-shot Flamingo-3B and achieve on-par performance with 4-shot Flamingo-80B.

4.4 Few-shot Video-Language Event Prediction

In this section, we show that our model not only can answer questions about the video visual features but also answering "What is more likely to happen next?". Given a video with associated subtitle transcript as premise, the video-language event prediction (VLEP) task is to predict the most likely future event. The original VLEP [23] paper formulates the problem as a binary classification problem where the model will be chosen from two possible future event candidates. Instead, we formulate this problem as another video-to-text generation problem to fit into our framework. Figure 5 depicts an

Table 3: Video QA results. BLIP$_{VQA}$ is finetuned on VQA [4]. ♠ indicates concurrent work. PT, FT indicates pretraining and finetuning.

Method	#video$_{PT}$	#video$_{FT}$	MSR-VTT	MSVD
BLIP	0	0-shot	0.55	0.45
BLIP	0	5-shot	0.84	0.53
BLIP$_{VQA}$ [26]	0	0-shot	19.2	35.2
VidIL(ours)	0	5-shot	**21.2**	**39.1**
♠Flamingo-3B [2]	27M	4-shot	14.9	33.0
♠Flamingo-3B [2]	27M	8-shot	19.6	37.0
♠Flamingo-80B [2]	27M	4-shot	23.9	41.7
♠Flamingo-80B [2]	27M	8-shot	27.6	45.5
ALPRO [25]	2M	full-shot	42.1	45.9

Table 4: Accuracy (%) on VLEP hidden test set.

Method	#video$_{FT}$	Acc
VLEP [23]	20142	67.5
MERLOT [66]	20142	68.4
VidIL(ours)	10-shot	**72.0**
Human	-	90.5

example with the same format as in Figure 2. Similar to the evaluation setting in QA, the generated free-form text will first be mapped to one of the two candidate answers using SentenceBert [45], and then calculate the accuracy. In Table 4, we report accuracy on the hidden test set of VLEP [23]. To our surprise, our 10-shot model beats state-of-the-art fully-supervised baseline, i.e., MERLOT [66], by a large margin ($\sim 4\%$). This is showing that our model has strong few-shot ability not only in video-language understanding but also in prediction. Since event prediction tasks rely heavily on temporal ordering, we show that with the proposed temporal-aware prompting, language models can be guided to capture temporal dynamics between historical and future events.

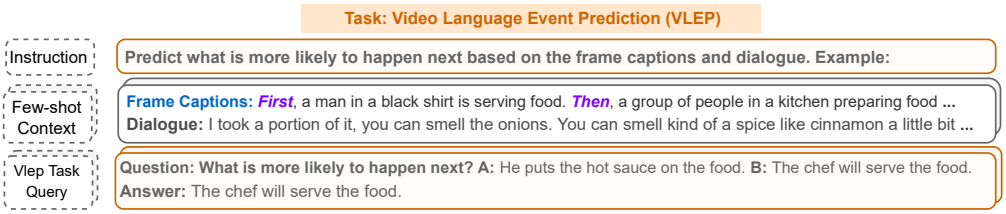

Figure 5: Prompt for VLEP task.

Table 5: Semi-supervised text-video retrieval with 10 labeled examples. V_{label} or $V_{unlabel}$ are the number of labeled and unlabeled videos, respectively. *t_R1* and *t_R* denote video-to-text Recall@1 and 5. *v_R1* and *v_R5* denote text-to-video Recall@1 and 5.

Model	Pseudo Label	MSR-VTT Retrieval					VaTex Retrieval				
		$V_{label}/V_{unlabel}$	t_R1	t_R5	v_R1	v_R5	$V_{label}/V_{unlabel}$	t_R1	t_R5	v_R1	v_R5
BLIP	-	-	33.2	57.2	40.5	62.8	-	28.2	53.4	**34.0**	58.6
BLIP	UniVL	10 / 7010	33.1	57.3	33.6	57.7	10 / 22685	25.5	47.7	26.1	49.1
BLIP	BLIP	10 / 7010	35.6	60.8	39.8	60.4	10 / 22685	26.3	50.5	29.3	53.6
BLIP	BLIP$_{cap}$	10 / 7010	35.3	58.0	39.1	63.3	10 / 22685	23.9	46.8	27.5	49.7
BLIP	VidIL(ours)	10 / 7010	**39.6**	**64.5**	**40.8**	**65.2**	10 / 22685	**33.3**	**59.1**	33.7	**59.5**
BLIP	Ground Truth	7010 / 0	43.6	66.2	43.1	67.2	22685 / 0	40.1	66.4	40.1	66.6
ALPRO [25]	Ground Truth	140200 / 0	32.0	60.6	33.9	60.7	-	-	-	-	-
DRL [54]	Ground Truth	180000 / 0	54.1	77.4	52.9	78.5	-	-	-	-	-

4.5 Semi-supervised Text-Video Retrieval

In addition to video-to-text generation tasks, we show that a broader range of video-language tasks can benefit from our few-shot video captioner from a data perspective. Here, we consider a low-budget semi-supervised setting where we only have a few labeled video-captain pairs and a large amount of unlabeled videos. The idea is to leverage our video captioner to generate **pseudo labels** for training any given vision-language models. As a case study, we evaluate on two text-video retrieval benchmarks, i.e., MSR-VTT and VaTeX. We use greedy decoding to generate pseudo caption for each video in the training set. We then train an identical base model, i.e., BLIP, using different pseudo labeled data as well as ground truth annotations. We report Recall @ 1 and 5 for both video-to-text

and text-to-video retrieval. Table 5 shows that through training on our pseudo labels, we can achieve significant improvements compared with zero-shot BLIP. We also show that the performance gain is not simply a result of training on more data, since finetuning on the pseudo labels generated by other baselines (UniVL, BLIP) is less effective and can even hurt the performance. Furthermore, on MSR-VTT Recall @ 5 we can even achieve comparable performance against BLIP model finetuned on full ground truth annotations.

Another interesting observation is that, compared with the video captioning results in Table 2, we found that the gain of our model over baselines on text-video retrieval is more visible than on captioning. A key factor in performing well on text-video retrieval tasks is to learn a good video-text multi-modal alignment. This result shows that our pseudo labels capture richer video-text alignment that can benefit the retrieval-style downstream task. The N-gram based generation metrics, e.g., BLEU, may not be able to fully reflect the alignment information, due to the variety of semantic levels in video captions. Furthermore, from a data perspective, our video captioner can be viewed as a data augmentation tool which is capable of generating or augmenting any open-domain video-language pretraining datasets with minimal human effort. As a result, we can potentially improving video-language pretraining by constructing a cleaner and more diverse video-text corpus.

Table 6: Impact of visual tokens.

Video Representation	Avg↑	Std↓
Frame	39.6	3.7
Frame+Object	40.3	2.9
Frame+Object+Event	39.9	2.8
Frame+Object+Event+Attribute	**40.8**	**2.4**

Table 7: Impact of the shot selection.

#shot	w/o in-context selection		w/ in-context selection	
	Avg↑	Std↓	Avg↑	Std↓
5-shot	38.4	2.1	40.4	1.2
10-shot	41.3	3.6	40.8	2.4
20-shot	42.6	3.3	42.2	2.0
30-shot	40.0	2.9	41.1	1.9

4.6 Ablation Studies

We perform comprehensive ablation studies on our few-shot prompt including the impact of different video representation, the number of shots and the in-context selection. All the ablation results are evaluated on MSVD_QA validation set, we report the mean and standard deviation of each setting on three sets of randomly sampled shots. For the cases **with** in-context example selection, we further select 5 examples as in-context examples from the sampled shots, while for the cases **without** in-context selection, all shots will be feed into the prompt. In Table 6, we show adding visual tokens consistently improves not only the model accuracy but also the model variance. A lower standard deviation indicates that the model is less sensitive to the few-shot sampling. In Table 7, we first show that, with the same context length, namely, 5 examples, in-context example selection significantly increases the performance as well as the robustness. At 10-shot, and 20-shot, directly fitting more shots into the prompt, i.e., w/o in-context selection, results in better performance than selecting from a larger support set while keeping the context length unchanged. Interestingly, at 30-shot, the situation reversed again. We observe that selecting 5 examples from a support set of size 30 results in better performance than adding all 30 shots as in-context example. This is showing that in-context selection can help our model utilize a larger number video examples. However, we still observe that the benefit of adding more shots saturated at around 20 to 30 shots, even if with in-context selection. we view this as a remaining challenging on how to make language models benefit from longer contexts.

5 Conclusions, Limitations and Future Work

This paper proposes VidIL, a few-shot **Vid**eo-language Learner via **I**mage and **L**anguage models. It demonstrates the strong ability of large-scale language models on performing video-to-text tasks when frame features are provided as unified text representations using image-language models. We propose a temporal order aware prompt by decomposing videos into a hierarchical structure, which is able to plug in multiple levels of frame features, along with speech transcripts. Without pretraining on videos, our model outperforms vision-language models learned from large-scale video datasets on a variety of few-shot tasks, such as domain-specific captioning, question answering, and future event prediction. One limitation is that the model is sensitive to the quality of in-context examples, as discussed in ablation studies. Future work will focus on leveraging large-scale language models to assist in video pretraining, such as its script knowledge for long video understanding and storytelling. For broader impact on the society, please refer to the supplemental material.

References

[1] Hassan Akbari, Liangzhe Yuan, Rui Qian, Wei-Hong Chuang, Shih-Fu Chang, Yin Cui, and Boqing Gong. Vatt: Transformers for multimodal self-supervised learning from raw video, audio and text. *Advances in Neural Information Processing Systems*, 34, 2021. 2

[2] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katie Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *arXiv preprint arXiv:2204.14198*, 2022. 3, 7, 8

[3] Jean-Baptiste Alayrac, Adria Recasens, Rosalia Schneider, Relja Arandjelović, Jason Ramapuram, Jeffrey De Fauw, Lucas Smaira, Sander Dieleman, and Andrew Zisserman. Self-supervised multimodal versatile networks. *Advances in Neural Information Processing Systems*, 33:25–37, 2020. 2

[4] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C. Lawrence Zitnick, and Devi Parikh. VQA: Visual Question Answering. In *International Conference on Computer Vision (ICCV)*, 2015. 7, 8

[5] Satanjeev Banerjee and Alon Lavie. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pages 65–72, 2005. 6

[6] 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. *Advances in neural information processing systems*, 33:1877–1901, 2020. 1, 2, 3, 4, 5

[7] David Chen and William Dolan. Collecting highly parallel data for paraphrase evaluation. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 190–200, Portland, Oregon, USA, June 2011. Association for Computational Linguistics. 6

[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] Jaemin Cho, Jie Lei, Hao Tan, and Mohit Bansal. Unifying vision-and-language tasks via text generation. In *International Conference on Machine Learning*, pages 1931–1942. PMLR, 2021. 3

[10] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 3

[11] Han Fang, Pengfei Xiong, Luhui Xu, and Yu Chen. Clip2video: Mastering video-text retrieval via image clip. *arXiv preprint arXiv:2106.11097*, 2021. 3

[12] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6202–6211, 2019. 3

[13] Tianyu Gao, Adam Fisch, and Danqi Chen. Making pre-trained language models better few-shot learners. *arXiv preprint arXiv:2012.15723*, 2020. 2

[14] Matt Gardner, Joel Grus, Mark Neumann, Oyvind Tafjord, Pradeep Dasigi, Nelson Liu, Matthew Peters, Michael Schmitz, and Luke Zettlemoyer. Allennlp: A deep semantic natural language processing platform. *arXiv preprint arXiv:1803.07640*, 2018. 2

[15] Ting-Hao Huang, Francis Ferraro, Nasrin Mostafazadeh, Ishan Misra, Aishwarya Agrawal, Jacob Devlin, Ross Girshick, Xiaodong He, Pushmeet Kohli, Dhruv Batra, et al. Visual storytelling. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1233–1239, 2016. 2

[16] Zhicheng Huang, Zhaoyang Zeng, Yupan Huang, Bei Liu, Dongmei Fu, and Jianlong Fu. Seeing out of the box: End-to-end pre-training for vision-language representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12976–12985, 2021. 2

[17] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc V Le, Yunhsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. *arXiv preprint arXiv:2102.05918*, 2021. 2

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

[19] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International journal of computer vision*, 123(1):32–73, 2017. 4

[20] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, et al. The open images dataset v4. *International Journal of Computer Vision*, 128(7):1956–1981, 2020. 4

[21] Jie Lei, Linjie Li, Luowei Zhou, Zhe Gan, Tamara L. Berg, Mohit Bansal, and Jingjing Liu. Less is more: Clipbert for video-and-language learningvia sparse sampling. In *CVPR*, 2021. 1, 3

[22] Jie Lei, Licheng Yu, Mohit Bansal, and Tamara L Berg. Tvqa: Localized, compositional video question answering. In *EMNLP*, 2018. 2

[23] Jie Lei, Licheng Yu, Tamara L Berg, and Mohit Bansal. What is more likely to happen next? video-and-language future event prediction. In *EMNLP*, 2020. 1, 6, 7, 8

[24] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*, 2019. 1

[25] Dongxu Li, Junnan Li, Hongdong Li, Juan Carlos Niebles, and Steven C.H. Hoi. Align and prompt: Video-and-language pre-training with entity prompts. In *arxiv*, 2021. 1, 2, 3, 8

[26] 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, 3, 6, 7, 8

[27] 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. *Advances in Neural Information Processing Systems*, 34, 2021. 2

[28] 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 *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2046–2065, 2020. 1, 2

[29] Xiujun Li, Xi Yin, Chunyuan Li, Pengchuan Zhang, Xiaowei Hu, Lei Zhang, Lijuan Wang, Houdong Hu, Li Dong, Furu Wei, et al. Oscar: Object-semantics aligned pre-training for vision-language tasks. In *European Conference on Computer Vision*, pages 121–137. Springer, 2020. 2

[30] Chin-Yew Lin and Franz Josef Och. Automatic evaluation of machine translation quality using longest common subsequence and skip-bigram statistics. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL-04)*, pages 605–612, 2004. 6

[31] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014. 6

[32] Xudong Lin, Gedas Bertasius, Jue Wang, Shih-Fu Chang, Devi Parikh, and Lorenzo Torresani. Vx2text: End-to-end learning of video-based text generation from multimodal inputs. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7001–7011. IEEE, 2021. 3

[33] Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. *arXiv preprint arXiv:1908.02265*, 2019. 2

[34] 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. 1, 2, 3, 6

[35] Huaishao Luo, Lei Ji, Ming Zhong, Yang Chen, Wen Lei, Nan Duan, and Tianrui Li. CLIP4Clip: An empirical study of clip for end to end video clip retrieval. *arXiv preprint arXiv:2104.08860*, 2021. 3

[36] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9879–9889, 2020. 2

[37] 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 *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2630–2640, 2019. 1, 2

[38] Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *arXiv preprint arXiv:2203.02155*, 2022. 2, 6

[39] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318, 2002. 6

[40] Mandela Patrick, Po-Yao Huang, Yuki Asano, Florian Metze, Alexander Hauptmann, Joao Henriques, and Andrea Vedaldi. Support-set bottlenecks for video-text representation learning. *arXiv preprint arXiv:2010.02824*, 2020. 2

[41] Baolin Peng, Chenguang Zhu, Chunyuan Li, Xiujun Li, Jinchao Li, Michael Zeng, and Jianfeng Gao. Few-shot natural language generation for task-oriented dialog. *arXiv preprint arXiv:2002.12328*, 2020. 1

[42] 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. *arXiv preprint arXiv:2103.00020*, 2021. 2, 3, 4

[43] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv preprint arXiv:1910.10683*, 2019. 1

[44] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21:1–67, 2020. 3

[45] Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, 2019. 4, 6, 8

[46] Timo Schick and Hinrich Schütze. Exploiting cloze questions for few shot text classification and natural language inference. *arXiv preprint arXiv:2001.07676*, 2020. 2

[47] Paul Hongsuck Seo, Arsha Nagrani, Anurag Arnab, and Cordelia Schmid. End-to-end generative pretraining for multimodal video captioning. *arXiv preprint arXiv:2201.08264*, 2022. 1

[48] Yixuan Su, Tian Lan, Yahui Liu, Fangyu Liu, Dani Yogatama, Yan Wang, Lingpeng Kong, and Nigel Collier. Language models can see: Plugging visual controls in text generation. *arXiv preprint arXiv:2205.02655*, 2022. 3

[49] Derek Tam, Rakesh R Menon, Mohit Bansal, Shashank Srivastava, and Colin Raffel. Improving and simplifying pattern exploiting training. *arXiv preprint arXiv:2103.11955*, 2021. 2

[50] Hao Tan and Mohit Bansal. Lxmert: Learning cross-modality encoder representations from transformers. *arXiv preprint arXiv:1908.07490*, 2019. 2

[51] Maria Tsimpoukelli, Jacob L Menick, Serkan Cabi, SM Eslami, Oriol Vinyals, and Felix Hill. Multimodal few-shot learning with frozen language models. *Advances in Neural Information Processing Systems*, 34:200–212, 2021. 3

[52] Ramakrishna Vedantam, C Lawrence Zitnick, and Devi Parikh. Cider: Consensus-based image description evaluation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4566–4575, 2015. 6

[53] Peng Wang, An Yang, Rui Men, Junyang Lin, Shuai Bai, Zhikang Li, Jianxin Ma, Chang Zhou, Jingren Zhou, and Hongxia Yang. Unifying architectures, tasks, and modalities through a simple sequence-to-sequence learning framework. *arXiv preprint arXiv:2202.03052*, 2022. 3

[54] Qiang Wang, Yanhao Zhang, Yun Zheng, Pan Pan, and Xian-Sheng Hua. Disentangled representation learning for text-video retrieval. *arXiv preprint arXiv:2203.07111*, 2022. 8

[55] Xin Wang, Jiawei Wu, Junkun Chen, Lei Li, Yuan-Fang Wang, and William Yang Wang. Vatex: A large-scale, high-quality multilingual dataset for video-and-language research. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 4580–4590. IEEE, 2019. 1, 6

[56] Zirui Wang, Jiahui Yu, Adams Wei Yu, Zihang Dai, Yulia Tsvetkov, and Yuan Cao. Simvlm: Simple visual language model pretraining with weak supervision. *arXiv preprint arXiv:2108.10904*, 2021. 2

[57] Jason Wei, Maarten Bosma, Vincent Y Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. Finetuned language models are zero-shot learners. *arXiv preprint arXiv:2109.01652*, 2021. 5

[58] 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 *ACM MM*, 2017. 1

[59] 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. *arXiv preprint arXiv:2109.14084*, 2021. 1

[60] Jun Xu, Tao Mei, Ting Yao, and Yong Rui. Msr-vtt: A large video description dataset for bridging video and language. In *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016. 1, 6

[61] Zhengyuan Yang, Zhe Gan, Jianfeng Wang, Xiaowei Hu, Yumao Lu, Zicheng Liu, and Lijuan Wang. An empirical study of gpt-3 for few-shot knowledge-based vqa. *arXiv preprint arXiv:2109.05014*, 2021. 1, 3

[62] Ziyi Yang, Yuwei Fang, Chenguang Zhu, Reid Pryzant, Dongdong Chen, Yu Shi, Yichong Xu, Yao Qian, Mei Gao, Yi-Ling Chen, et al. i-code: An integrative and composable multimodal learning framework. *arXiv preprint arXiv:2205.01818*, 2022. 1

[63] Fei Yu, Jiji Tang, Weichong Yin, Yu Sun, Hao Tian, Hua Wu, and Haifeng Wang. Ernie-vil: Knowledge enhanced vision-language representations through scene graphs. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 3208–3216, 2021. 2

[64] Lu Yuan, Dongdong Chen, Yi-Ling Chen, Noel Codella, Xiyang Dai, Jianfeng Gao, Houdong Hu, Xuedong Huang, Boxin Li, Chunyuan Li, et al. Florence: A new foundation model for computer vision. *arXiv preprint arXiv:2111.11432*, 2021. 2

[65] 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. *arXiv preprint arXiv:2201.02639*, 2022. 1

[66] Rowan Zellers, Ximing Lu, Jack Hessel, Youngjae Yu, Jae Sung Park, Jize Cao, Ali Farhadi, and Yejin Choi. Merlot: Multimodal neural script knowledge models. *Advances in Neural Information Processing Systems*, 34, 2021. 1, 8

[67] Andy Zeng, Adrian Wong, Stefan Welker, Krzysztof Choromanski, Federico Tombari, Aveek Purohit, Michael Ryoo, Vikas Sindhwani, Johnny Lee, Vincent Vanhoucke, et al. Socratic models: Composing zero-shot multimodal reasoning with language. *arXiv preprint arXiv:2204.00598*, 2022. 3, 4

[68] Pengchuan Zhang, Xiujun Li, Xiaowei Hu, Jianwei Yang, Lei Zhang, Lijuan Wang, Yejin Choi, and Jianfeng Gao. Vinvl: Revisiting visual representations in vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5579–5588, 2021. 2

[69] Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, et al. Opt: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068*, 2022. 1

[70] Zihao Zhao, Eric Wallace, Shi Feng, Dan Klein, and Sameer Singh. Calibrate before use: Improving few-shot performance of language models. In *International Conference on Machine Learning*, pages 12697–12706. PMLR, 2021. 1

[71] Luowei Zhou, Hamid Palangi, Lei Zhang, Houdong Hu, Jason Corso, and Jianfeng Gao. Unified vision-language pre-training for image captioning and vqa. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 13041–13049, 2020. 2

[72] Luowei Zhou, Chenliang Xu, and Jason J Corso. Towards automatic learning of procedures from web instructional videos. In *AAAI Conference on Artificial Intelligence*, pages 7590–7598, 2018. 6

[73] Xizhou Zhu, Jinguo Zhu, Hao Li, Xiaoshi Wu, Xiaogang Wang, Hongsheng Li, Xiaohua Wang, and Jifeng Dai. Uni-perceiver: Pre-training unified architecture for generic perception for zero-shot and few-shot tasks. *arXiv preprint arXiv:2112.01522*, 2021. 3

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] See Section 4.6.
   (c) Did you discuss any potential negative societal impacts of your work? [Yes] See supplemental material.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [N/A]
   (b) Did you include complete proofs of all theoretical results? [N/A]
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] See supplemental material.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See Section 4.1 and supplemental material.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] See Section 4.6.
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] See Section 4.1.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes] see Section 4.6.
   (b) Did you mention the license of the assets? [Yes] see the citations in Section 4.6.
   (c) Did you include any new assets either in the supplemental material or as a URL? [N/A]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [Yes] see the citations in Section 4.6.
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [Yes] see the citations in Section 4.6.
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]