Panda-70M: Captioning 70M Videos with Multiple Cross-Modality Teachers

Tsai-Shien Chen[1,2,*] Aliaksandr Siarohin[1] Willi Menapace[1,3,*] Ekaterina Deyneka[1]
Hsiang-wei Chao[1] Byung Eun Jeon[1] Yuwei Fang[1] Hsin-Ying Lee[1] Jian Ren[1]
Ming-Hsuan Yang[2] Sergey Tulyakov[1]
[1]Snap Inc. [2]University of California, Merced [3]University of Trento

https://snap-research.github.io/Panda-70M

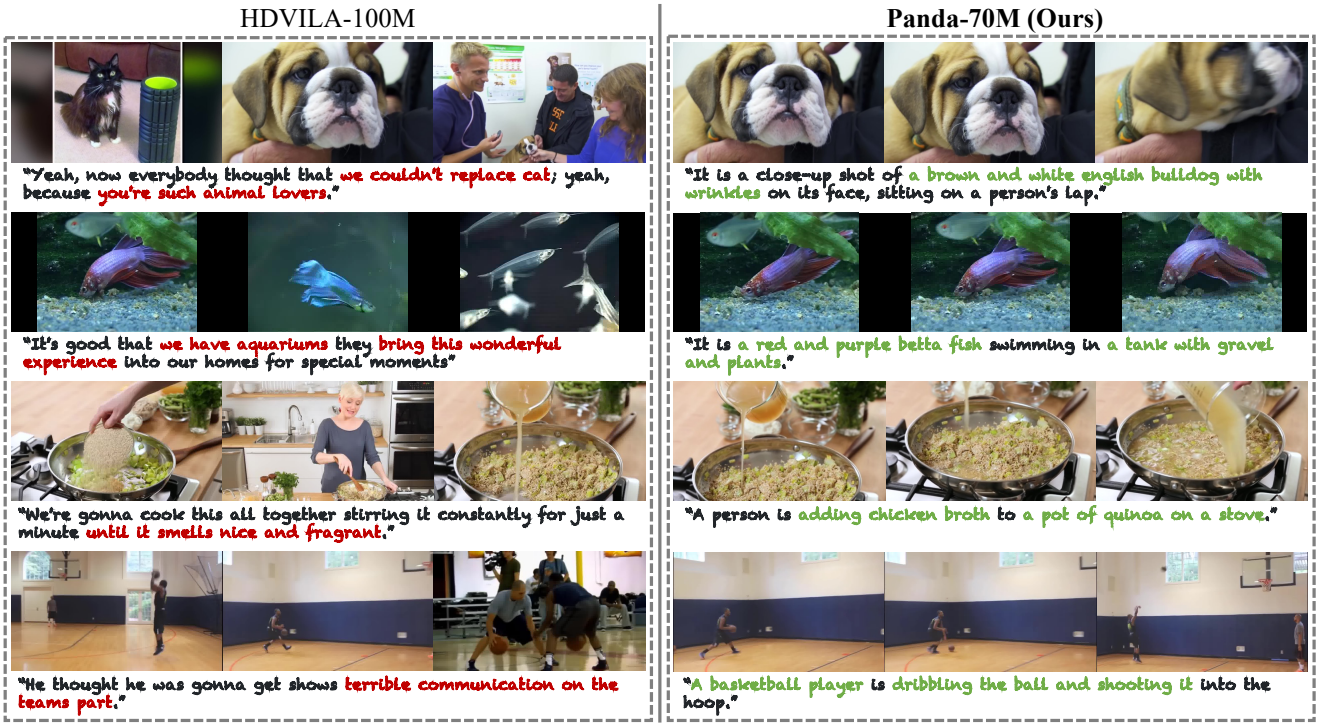

Figure 1. **Comparison of Panda-70M to the existing large-scale video-language datasets.** We introduce Panda-70M, a large-scale video dataset with captions that are annotated by multiple cross-modality vision-language models. Compared to text annotations in existing dataset [80], captions in Panda-70M more precisely describe the main object and action in videos (highlighted in green). Besides, videos in Panda-70M are semantically coherent, high-resolution, and free from watermarks. More samples can be found in Appendix E.

Abstract

The quality of the data and annotation upper-bounds the quality of a downstream model. While there exist large text corpora and image-text pairs, high-quality video-text data is much harder to collect. First of all, manual labeling is more time-consuming, as it requires an annotator to watch an entire video. Second, videos have a temporal dimension, consisting of several scenes stacked together, and showing multiple actions. Accordingly, to establish a video dataset with high-quality captions, we propose an automatic approach leveraging multimodal inputs, such as textual video description, subtitles, and individual video frames. Specifically, we curate 3.8M high-resolution videos from the publicly available HD-VILA-100M dataset. We then split them into semantically consistent video clips, and apply multiple cross-modality teacher models to obtain captions for each video. Next, we finetune a retrieval model on a small subset where the best caption of each video is manually selected and then employ the model in the whole dataset to select the best caption as the annotation. In this way, we get 70M videos paired with high-quality text captions. We dub the dataset as Panda-70M. We show the value of the proposed dataset on three downstream tasks: video captioning, video and text retrieval, and text-driven video generation. The models trained on the proposed data score substantially better on the majority of metrics across all the tasks.

[*] This work was done while interning at Snap.

Table 1. **Comparison of Panda-70M and other video-language datasets.** We split the datasets into two groups: the group at the top is annotated by ASR, and the group at the bottom is labeled with captions.

Dataset	Year	Text	Domain	#Videos	Avg/Total video len		Avg text len	Resolution
HowTo100M [52]	2019	ASR	Open	136M	3.6s	134.5Khr	4.0 words	240p
ACAV [32]	2021	ASR	Open	100M	10.0s	277.7Khr	-	-
YT-Temporal-180M [87]	2021	ASR	Open	180M	-	-	-	-
HD-VILA-100M [80]	2022	ASR	Open	103M	13.4s	371.5Khr	32.5 words	720p
MSVD [13]	2011	Manual caption	Open	1970	9.7s	5.3h	8.7 words	-
LSMDC [58]	2015	Manual caption	Movie	118K	4.8s	158h	7.0 words	1080p
MSR-VTT [79]	2016	Manual caption	Open	10K	15.0s	40h	9.3 words	240p
DiDeMo [3]	2017	Manual caption	Flickr	27K	6.9s	87h	8.0 words	-
ActivityNet [11]	2017	Manual caption	Action	100K	36.0s	849h	13.5 words	-
YouCook2 [93]	2018	Manual caption	Cooking	14K	19.6s	176h	8.8 words	-
VATEX [73]	2019	Manual caption	Open	41K	$\sim$10s	$\sim$115h	15.2 words	-
Panda-70M (Ours)	2024	Automatic caption	Open	70.8M	8.5s	166.8Khr	13.2 words	720p

1. Introduction

We enter an era where the size of computing and data are indispensable for large-scale multimodal learning. Most breakthroughs are achieved by large-scale computing infrastructure, large-scale models, and large-scale data. Due to these integral components, we have powerful text-to-image [4, 57, 59, 61, 83] and image-to-text models [2, 36, 43, 53]. Scaling the model size or the compute is challenging and expensive; however, it requires a finite amount of engineering time. Scaling the data is relatively more challenging, as it takes time for a human to analyze each sample.

Especially, compared to image-text pairs [10, 12, 62], video-text pairs are even harder to obtain. First, annotating videos is more time-consuming, as an annotator needs to watch the entire video before labeling. Second, videos often contain multiple scenes stitched together and consist of temporally varying content. Finally, meta-information, such as subtitles, video description, and voice-over, is often too broad or not correctly aligned in time or cannot precisely describe a video. For example, several 100M-scale datasets, such as HD-VILA-100M [80] and HowTo100M [52], are annotated by automatic speech recognition (ASR). However, as shown in Figure 1, the subtitles usually fail to include the main content and action presented in the video. This limits the value of such datasets for multimodal training. We summarize the datasets available to the community in Table 1. Some are low-resolution, some are annotated by ASR, some contain data from a limited domain, some are small-scale, and some offer short captions.

In this work, we present a large-scale dataset containing 70M video clips with caption annotations. It includes high-resolution videos from an open domain with rich captions averaging 13.2 words per caption. While manually annotating 70M videos is prohibitively expensive, we opt for automatic annotation. Our key insight is that a video typically comes with information from several modalities that can assist automatic captioning. This includes the title, description, subtitles of the video, individual static frames, and the video itself. The value of this data cannot be fully maximized when only partially used. In comparison, we propose to utilize different combinations of multimodal data as inputs to various cross-modality captioning models. To substantiate this idea, we conduct a numerical analysis based on a human evaluation (the details are provided in Appendix B.3). If we use multiple cross-modality models to caption some video samples and evaluate the results by showing them to humans, we see that there is no single model able to generate good captions for more than 31% of videos. However, if we jointly collect all the captions from different models, we observe that 84.7% of videos can be annotated with at least one good caption.

To establish the dataset with this mindset, we begin by using 3.8M high-resolution long videos collected from HD-VILA-100M [80] and process them through the following three steps. First, we design a semantics-aware video splitting algorithm to cut long videos into semantically consistent clips while striking the balance between semantics coherence and the duration of the video clips. Second, we use a range of cross-modality teacher models, including image captioning models [37] and image/video visual-question-answering (VQA) models [38, 88, 94] with additional text inputs, such as video description and subtitles, to predict several candidate captions for a clip. Lastly, we collect a 100K video subset, where human annotators act as an oracle to select the best caption for each video. We use this dataset to finetune a fine-grained video-to-text retrieval model [39] which is then applied to the whole dataset to select the most precise caption as the annotation. Running multiple teacher models is computationally expensive and time-consuming. To pursue efficient video captioning at scale in the future, we train a student model to distill the knowledge from the teachers. The student model adopts a two-branch architecture which can take both visual and textual inputs to benefit the captioning from multimodal information.

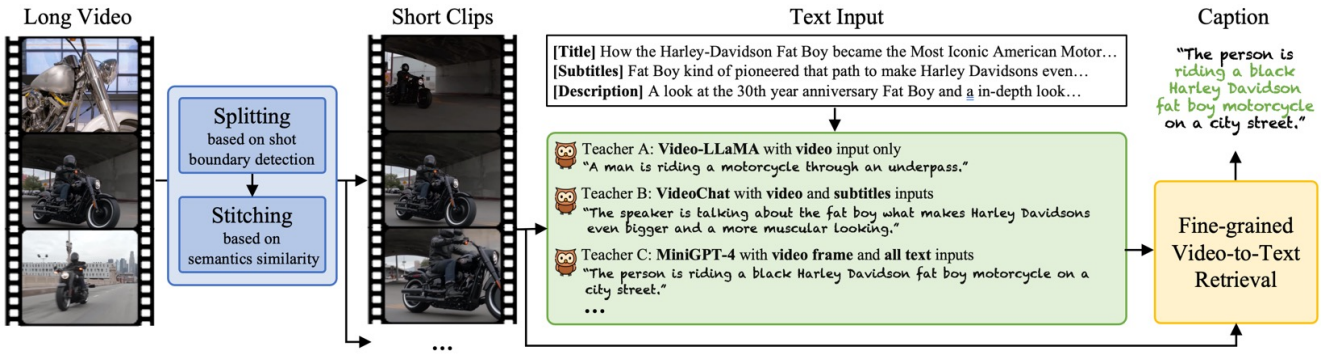

Figure 2. **Video captioning pipeline.** Given a long video, we first split it into several semantically coherent clips. Subsequently, we utilize a number of teacher models with different multimodal inputs to generate multiple captions for a video clip. Lastly, we finetune a fine-grained retrieval model to select the caption that best describes the video clip as the annotation.

Extensive experiments demonstrate that pretraining with the proposed Panda-70M[1] can benefit several downstream tasks, including video captioning, video and text retrieval, and text-to-video generation. We also show that training a student model in a knowledge distillation manner facilitates learning a strong student model which can outperform any teacher model by more than 7.7% preference ratio as in Table 3, where the performance can be further enhanced by additional text inputs, like video description and subtitles.

2. Related Work

Vision-Language Datasets. Training with millions or even billions of image-text pairs [10, 31, 55, 62, 86] has been shown to be effective in learning powerful image foundation models [2, 6, 21, 25, 27, 82]. With this work, our goal is to build a large video-language dataset containing rich captions. We compare related datasets in Table 1. Several precedent video-language datasets [3, 11, 13, 58, 73, 79, 93] contain data tackling various tasks, such as action recognition, video understanding, VQA, and retrieval. However, manually annotating data is costly and limits the scale of such datasets (typically they contain less than 120K samples). To alleviate the lack of data, the works of [52, 80, 87] propose to automatically annotate data with subtitles, generated by ASR. While this approach significantly increases the dataset scale reaching 100M of samples, the subtitles, unfortunately, do not precisely describe the main video content, as shown in Figure 1. In comparison, in this work, we propose an automatic captioning pipeline with the inputs of multimodal data that enables us to scale up the dataset of high-quality video-caption pairs to a 70M scale.

Vision-Language Models learn the correlation between visual data (images or videos) and linguistic signals (words or sentences) and can be applied to several downstream applications, including text-driven image or video generation [4, 8, 29, 57, 59, 61, 65, 83, 92], captioning [2, 36, 37, 43, 63, 81], VQA [14, 38, 53, 88, 94] and retrieval [17, 39, 49]. We utilize several vision-language models for the annotation of Panda-70M. BLIP-2 [37] introduces an efficient vision-language pretraining that can facilitate image captioning. We use BLIP-2 as one of the teachers and input a randomly sampled video frame for captioning. MiniGPT-4 [94] is an image VQA model that learns a projection layer to align a large language model (LLM) and a visual encoder. In addition to a video frame, we also input a prompt with extra text information, such as video description and subtitles, and ask the model to summarize all multimodal inputs. For the video modality, Video-LLaMA [88] and VideoChat [38] are both video VQA models and learn to extract LLM-compatible visual embeddings. We use both models and ask them to caption a video with prompt input. Besides, Unmasked Teacher [39] is a video foundation model which can facilitate video understanding. We finetune it to implement fine-grained retrieval and use it to select the more precise caption as the annotation.

Video Annotation through Multi-modal Models. With the aforementioned development on vision-language models, some concurrent works [7, 72, 75] also leverage these models for video captioning. VideoFactory [72] employs BLIP-2 [37] to caption video clips. However, as reported in Appendix B.3, the performance of a single BLIP-2 model is suboptimal. More similar to our captioning pipeline, InternVid [75] and Stable Video Diffusion [7] also use multiple captioning models which are followed by an LLM for summarization. In practice, we found the LLM would propagate errors from noisy outputs of vision-language models.

3. Methodology

To build Panda-70M, we utilize 3.8M high-resolution long videos collected from HD-VILA-100M [80]. We then split them into 70.8M semantically coherent clips as described in Section 3.1. Section 3.2 shows how multiple cross-modality teacher models are used to generate a set of candidate caption annotations. Next, we finetune a fine-grained retrieval model to select the most accurate caption as detailed in Section 3.3. Finally, in Section 3.4, we describe our approach to

[1]We call our dataset Panda, drawing an analogy to Panda Po, who learns from multiple martial arts teachers.

training a student captioning model using Panda-70M. The high-level view of our approach is shown in Figure 2.

3.1. Semantics-aware Video Splitting

A desired video sample in a video-captioning dataset should have two somewhat contradictory characteristics. On the one hand, the video should be semantically consistent, so the video samples can better benefit the downstream tasks, such as action recognition, and the caption can also more accurately express its semantics content without ambiguity. On the other hand, the video cannot be too short or fragmentary to contain meaningful motion content, which is beneficial to tasks, like video generation.

To achieve both goals, we design a two-stage semantics-aware splitting algorithm to cut a long video into semantically coherent clips. In the first stage, we split the video based on shot boundary detection [1], as the semantics often change when a new scene starts. In the second stage, we stitch adjacent clips if they are incorrectly separated by the first stage, ensuring the videos do not end up being too short. To do so, we use ImageBind [25] to extract embeddings of video frames and merge the adjacent clips if the frame embeddings from two clips are similar. We also implement additional procedures to handle: 1) long videos without any cut-scenes, 2) videos using complex transitions, such as fade-in and fade-out effects, which are not usually detected as cut-scenes, and 3) removal of redundant clips to increase the diversity of the dataset. More details of the splitting algorithm are in Appendix A. Notably, while our dataset focuses on fine-grained video-text pairs with consistent semantics, users can still acquire long videos with multiple cut-scenes by concatenating consecutive clips and captions, as these clips are split from the same long video.

To quantitatively verify the semantic consistency of a video clip, we introduce Max Running LPIPS, which highlights the most significant perceptual change within a video clip. Formally, given an n-second video clip, we subsample the video frames each second and denote the keyframes as $\{f_1, ..., f_n\}$. The Max Running LPIPS is formulated as:

$$\max(\{\text{LPIPS}(f_i, f_{i+1}) \mid i \in [1, n-1]\}). \quad (1)$$

where $\text{LPIPS}(\cdot, \cdot)$ is the perceptual similarity [89] of two images. As in Table 2, our splitting achieves a better semantics consistency than the splitting based on the alignment of subtitles sentences [52, 80], while maintaining longer video length than the vanilla shot boundary detection [1].

3.2. Captioning with Cross-Modality Teachers

Videos in HD-VILA-100M [80] contain rich multimodal information beneficial for captioning. Specifically, besides the video itself, there are also useful texts (*e.g.*, video title, description, and subtitles) and images (*e.g.*, individual video frames). Driven by this insight, we propose to use several captioning models with the inputs of different modalities.

Table 2. **Comparison of splitting algorithms.** We split 1K long videos by three algorithms and test the semantics consistency of the output clips by the proposed Max Running LPIPS. Our splitting strikes a better balance for the trade-off between semantics consistency and clip length.

Method	Max running LPIPS↓	Avg Video Len
Sub. Align [52, 80]	0.408	11.8s
PySceneDetect [1]	0.247	4.1s
Our Splitting	0.256	7.9s

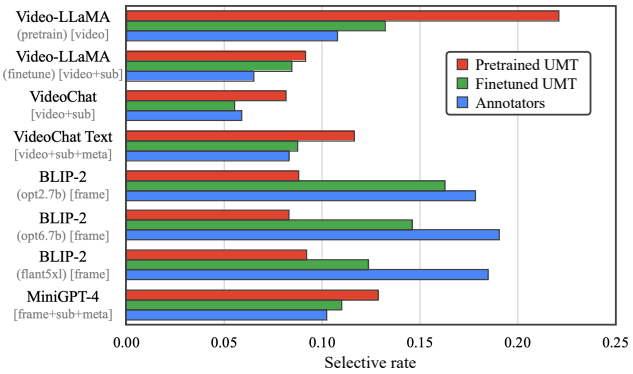

Figure 3. **Distributions of the selective rate of teacher models.** We plot the distributions of the selective rate of eight teachers on 1,805 testing videos. The results are based on the selection of the pretrained (red) or finetuned (green) Unmasked Teacher [39] and human annotators (blue).

We start with a large pool including 31 captioning models. The introduction of the model pool is in Appendix B.1. Since running the inference of all models on 70M video clips is computationally expensive, we construct a short list of eight well-performing models based on a user study. The list is shown in the y-axis of Figure 3. More details of this process are in Appendix B.3. Briefly, the models are composed of five base models with different pretraining weights and input information. The five base models include Video-LLaMA [88] (video VQA), VideoChat [38] (video VQA), VideoChat Text [38] (natural language model which textualizes the video content), BLIP-2 [37] (image captioning), and MiniGPT-4 [94] (image VQA). To implement video captioning by cross-modality teacher models, we formulate distinct captioning processes tailored to each modality. For example, for the VQA models, in addition to visual data, we also input a prompt with additional text information and ask the models to summarize all multimodal inputs into one sentence. Details on the captioning process of each teacher model are described in Appendix B.2.

We hypothesize that teacher models using different modality data perform well on different kinds of videos. For example, video models can perform better on videos with complex dynamics due to the additional modules to handle temporal information. On the other hand, image models can accurately caption the videos with rare and uncommon objects, since they were trained using large-scale datasets of

image-text pairs [62]. Finally, for videos that are visually hard to understand, VQA models have leverage as they can employ additional textual clues.

This hypothesis can be supported by a numerical evaluation. Specifically, we conduct a user study where the participants are asked to select the best caption from eight candidates. We plot the selective rate of each teacher model in Figure 3 (blue bars). The results show that the best captions are generated by different teacher models. Moreover, the highest selective rate of an individual teacher model (*i.e.*, BLIP-2 with opt6.7b [90]) is only 17.85%. This fact expresses the limited captioning capability of a single model on a wide variety of videos.

3.3. Fine-grained Video-to-Text Retrieval

Given multiple candidate captions for a video, we seek the one that best aligns with the video content. An intuitive idea is to use the available generic video-to-text retrieval models [25, 39] to pick such a caption. Unfortunately, we find that they usually fail to pick the optimal result. One reason is that generic models are trained using contrastive learning objectives [15, 82] and learn to distinguish one sample from other completely unrelated samples[2]. In contrast, in our case, all candidate captions are highly relevant to the video sample and require the model to discern subtle distinctions within each caption for optimal performance.

To tailor the retrieval model to our "fine-grained" retrieval scenario, we collect a subset of 100K videos, for which human annotators select the caption containing the most correct and detailed information about the main content of the video. We then finetune Unmasked Teacher [39] (UMT) on this dataset. We implement hard negative mining [16, 35] for contrastive loss, where the seven captions not selected by annotators compose the hard negative samples and are assigned a larger training weight. We describe the details of the dataset collection and finetuning of UMT in Appendix C.1 and C.2 respectively.

We quantitatively evaluate the retrieval performance of UMTs with and without finetuning on the validation set. The experiments indicate that a finetuned UMT can achieve 35.90% R@1 accuracy which significantly outperforms a pretrained UMT which has 21.82% R@1. Notably, we conducted a human agreement evaluation by asking two other persons to re-perform the annotation and comparing the results with the original annotations. The average human agreement score is only 44.9% R@1 showing that the task is subjective when more than one caption is equally good. Alternatively, if we consider the captions selected by any of the three persons as good captions (*i.e.*, a video might have multiple good captions), UMT achieves 78.9% R@1. Besides, in Figure 3, we show that a finetuned UMT (green bars) can select the captions distributed similarly to human-selected captions (blue bars). We run the finetuned UMT on the whole dataset to select the best caption as the annotation as elaborated in Appendix C.3.

[2]Negative samples for contrastive learning [15] are usually randomly sampled from the within-batch data and show no association to the anchor.

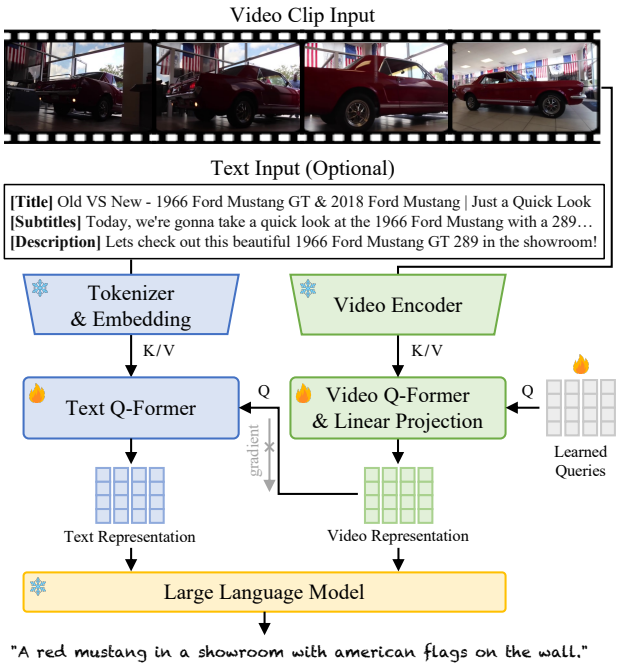

Figure 4. **Architecture of student captioning model.**

3.4. Multimodal Student Captioning Model

While the aforementioned captioning pipeline can generate promising captions, the heavy computational demands hinder its capability to expand the dataset to an even larger scale. Indeed, one needs to run $8+1$ different models to annotate a single video clip. To deal with this problem, we learn a student captioning model on Panda-70M to distill the knowledge from multiple teacher models.

As shown in Figure 4, the student model includes visual and text branches, leveraging multimodal inputs. For the vision branch, we use the same architecture as Video-LLaMA [88] to extract LLM-compatible video representation. For the text branch, a straightforward design is to directly input text embedding into the LLM. However, this will lead to two problems: first, the text prompt with video description and subtitles can be too long, dominating the decision of the LLM and burdening heavy computation; second, the information from the description and subtitles is often noisy and not necessary to align with the content of the video. To tackle this, we add a text Q-former to extract the text representation with fixed length and better bridge the video and text representations. The Q-former has the same architecture as the Query Transformer in BLIP-2 [37]. During training, we block the gradient propagation from the text branch to the vision branch and train the visual encoder only based on the video input. More details about the architecture and training of the student model are in Appendix D.

Table 3. **Zero-shot video captioning (%).** We compare Video-LLaMA [88] with official weight (pretrained on 2.5M videos and 595K images) and our Panda-2M pretraining weight. We also test our student model (with vision branch only) trained on the complete Panda-70M dataset. We report BLEU-4 (B-4) [54], ROUGE-L (R) [40], METEOR (M) [5], CIDEr (C) [69], and BERTScore (BERT) [91] on two benchmarks MSR-VTT [79] and MSVD [13].

Method	Pretraining Data	MSR-VTT					MSVD				
		B4↑	R↑	M↑	C↑	BERT↑	B4↑	R↑	M↑	C↑	BERT↑
Video-LLaMA [88]	2.5M vid + 595K img	5.8	30.0	15.9	14.3	84.5	12.7	43.0	23.6	38.5	87.3
Video-LLaMA [88]	Panda-2M (Ours)	23.5	48.6	26.7	29.1	87.2	31.2	59.9	34.7	47.0	89.8
Student (Ours)	**Panda-70M (Ours)**	**25.4**	**50.1**	**27.7**	**31.5**	**87.9**	**32.8**	**61.2**	**35.3**	**49.2**	**90.2**

Table 4. **Comparison of the teacher(s) and student captioning models (%).** We conduct a user study to compare single teacher, all teacher, and two student models (with and without text).

Model	Preference Ratio↑
Video-LLaMA [88] (pretrain)	9.4
Video-LLaMA [88] (finetune)	7.0
VideoChat [38]	7.7
VideoChat Text [38]	3.3
BLIP-2 [37] (opt2.7b)	10.7
BLIP-2 [37] (opt6.7b)	9.0
BLIP-2 [37] (flant5xl)	9.9
MiniGPT-4 [94]	3.1
Student (video input) (Ours)	18.4
Student (video+text inputs) (Ours)	21.4
All Teachers (Ours)	**23.3**

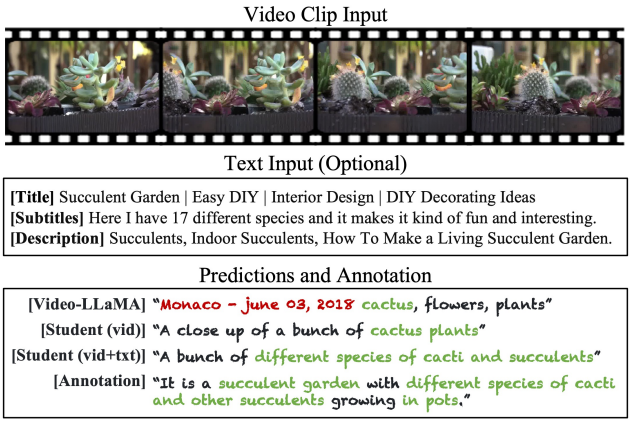

Figure 5. **Qualitative comparison of video captioning.** We visualize a sample from the testing set of Panda-70M and show its annotation (bottommost). We also show the captions predicted from three models, including Video-LLaMA [88] with official weight and the student models with video-only or video and text inputs.

4. Experiments

We visualize the samples of Panda-70M in Appendix E. To quantitatively evaluate the effectiveness of Panda-70M, we test its pretraining performance on three downstream applications: video captioning in Section 4.1, video and text retrieval in Section 4.2, and video generation in Section 4.3. The training details of the downstream models adhere to the official codebases unless explicitly specified.

4.1. Video Captioning

Experiment setup. To evaluate the performance of video captioning, we use Video-LLaMA [88] with the vision branch only as the base model. We compare two pretraining weights: the official weight, which is jointly trained on 2.5M video-text pairs and 595K image-text pairs [43], and the weight trained on our Panda-2M from scratch. Panda-2M is a randomly sampled subset of Panda-70M and shares the same amount of training samples as the official weight. We also train our student model with both video and text branches on complete Panda-70M for better captioning performance. For all models, we use the same backbone, using Vicuna-7B [18] as the large-language model, ViT [22] and Q-Former [37] as the video encoder, and the linear projection layer from MiniGPT-4 [94]. For Panda-2M pretraining, we only use the video and caption data without using other textual information for a fair comparison. For the student model, in addition to the video, we also randomly input the metadata and subtitles into the model during training.

Downstream datasets and evaluation metrics. We test zero-shot video captioning on two benchmarks: MSR-VTT [79] and MSVD [13]. MSR-VTT contains 10K videos with 20 manually annotated captions for each video; we report the results on the 2,990 testing split. MSVD consists of 1,970 videos with a total of 80K descriptions; we report the numbers on the 670 testing videos. Note that we do not use any training or validation videos from the downstream datasets. To quantitatively evaluate the quality of output captions, we follow the common protocols [41, 48, 78] and report BLEU-4 [54], ROGUE-L [40], METEOR [5], and CIDEr [69]. All the metrics are computed using the pycocoevalcap [42] package. We also compute BERTScore [91] to evaluate the contextual similarity for each token in the ground truth and the predicted captions. The results are reported in Table 3. For a fair comparison, we do not input any additional text information to the student model during the inference on the downstream datasets. In Figure 5, we also showcase a video sample from the testing set of Panda-70M and the predicted captions for the qualitative comparison.

Table 5. **Video and text retrieval (%).** We compare the Unmasked Teacher [39] with the official checkpoint (pretrained on 2.5M videos and 3M images) and our Panda-5M pretraining. We evaluate their performance on zero-shot and finetune text-to-video (T2V) and video-to-text (V2T) retrieval. We report R@1, R@5, and R@10 accuracy on three benchmarks: MSR-VTT [79], DiDeMo [3], and MSVD [13].

Method	Pretraining Data	MSR-VTT			DiDeMo			MSVD		
		R@1↑	R@5↑	R@10↑	R@1↑	R@5↑	R@10↑	R@1↑	R@5↑	R@10↑
Zero-shot T2V / V2T Retrieval										
AlignPrompt [34]	2.5M vid + 3M img	24.1 / -	44.7 / -	55.4 / -	23.8 / -	47.3 / -	57.9 / -	- / -	- / -	- / -
BridgeFormer [24]	2.5M vid + 3M img	26.0 / -	46.4 / -	56.4 / -	25.6 / -	50.6 / -	61.6 / -	43.6 / -	74.9 / -	84.9 / -
UMT [39]	2.5M vid + 3M img	30.2 / 33.3	51.3 / 58.1	61.6 / 66.7	33.6 / 32.1	58.1 / 57.3	65.5 / **66.7**	66.3 / **44.4**	85.5 / **73.3**	89.3 / **82.4**
UMT [39]	**Panda-5M (Ours)**	**37.2 / 36.3**	**58.1 / 61.0**	**69.5 / 69.7**	**34.2 / 33.4**	**58.4 / 57.9**	**66.5** / 65.8	**71.2** / 37.2	**88.4** / 65.1	**92.7** / 75.6
Finetune T2V / V2T Retrieval										
CLIP4Clip [49]	400M img	44.5 / 40.6	71.4 / 69.5	81.6 / 79.5	43.4 / 42.5	70.2 / 70.6	80.6 / 80.2	46.2 / 62.0	76.1 / 87.3	84.6 / 92.6
X-CLIP [50]	400M img	49.3 / 48.9	75.8 / 76.8	84.8 / 84.5	50.4 / **66.8**	80.6 / **90.4**	- / -	47.8 / 47.8	79.3 / 76.8	- / -
InternVideo [74]	146M vid + 100M img	55.2 / 57.9	- / -	- / -	57.9 / 59.1	- / -	- / -	**58.4** / 76.3	- / -	- / -
UMT [39]	2.5M vid + 3M img	53.3 / 51.4	76.6 / 76.3	83.9 / 82.8	59.7 / 59.5	84.9 / 84.5	90.8 / **90.7**	53.7 / 77.2	80.5 / 91.6	86.8 / 94.8
UMT [39]	**Panda-5M (Ours)**	**58.4 / 58.5**	**80.9 / 81.0**	**86.9 / 87.0**	**60.6** / 58.9	**86.0** / 84.6	**92.4** / 90.4	57.5 / **81.3**	**83.6 / 93.7**	**89.5 / 96.6**

As in Table 3, Video-LLaMA with Panda-2M pretraining weight achieves significantly superior performance compared to the official weight. Numerically, our pretraining weight yields 17.7% and 18.5% improvement respectively on MSR-VTT and MSVD in terms of B-4. Besides, in Figure 5, we can find that the caption from the original Video-LLaMA contains irrelevant and generic information, such as date and location. In comparison, our prediction better aligns with the video content.

Can the student perform better than its teacher? In Section 3.4, we learn a student model in a knowledge distillation manner. To evaluate the performance of the student model, we conduct a user study where participants are asked to select the best caption from ten candidates for each video. Ten captions are predicted from eight teacher models and two student models (with and without text inputs). We collect the results from five participants to reduce the personal subjective bias. Each participant saw the same 200 videos, which were randomly sampled from the testing set and had not been seen during the training of the student model and UMT. We report the preference ratio of each model and the R@1 accuracy of the finetuned UMT (*i.e.*, all teachers) in Table 4. We can observe that the student model outperforms any individual teacher model and achieves a comparable performance with all teacher models.

Can multimodal inputs leverage video captioning? Our student model supports both video and text inputs. In Table 4, we show that the student model with both video and text inputs outperforms the model with video input only by 3.0% preference ratio. Qualitatively, we show the predictions with and without text inputs in Figure 5. While the prediction with pure video input can include partial content of the video, like "cactus", the model with both video and text inputs can more comprehensively include keywords such as "succulents" and "different species" from the video title, description, and subtitles.

4.2. Video and Text Retrieval

Experiment setup. We use Unmasked Teacher [39] as the base model to evaluate the performance on video and text retrieval. The standard protocols [17, 24, 34, 39, 78] jointly use 3M images from CC3M [64] and 2.5M videos as the pretraining datasets. Thus, we randomly sample a Panda-5M subset, which shares the same number of training samples as the standard pretraining dataset for a fair comparison. For both datasets, we use the same backbone composed of ViT-L/16 [21] and BERTlarge [20]. We use the official weights for the standard datasets pretraining and train the model from scratch for our Panda-5M.

Downstream datasets and evaluation metric. We test both zero-shot and finetune retrieval on three benchmarks: MSR-VTT [79], DiDeMo [3], and MSVD [13]. For MSR-VTT, we follow the common protocol [34, 85] to evaluate on 1K testing split, which is not the same as the testing videos for captioning in Section 4.1. For DiDeMo [3], it contains 10K Flickr videos with a total of 40K dense captions. As in the previous standard [24, 33, 44], we evaluate paragraph-to-video retrieval by concatenating all sentence descriptions of one video into a single query. We report the results on the 1K testing set. For MSVD [13], we report the results on the 670 testing videos. We employ the standard metric and report R@1, R@5, and R@10 accuracy on both text-to-video and video-to-text retrieval in Table 5.

We can observe that pretraining with our Panda-5M outperforms the official weight in both zero-shot and finetune retrieval settings. Especially, our pretraining yields 7.0%, 0.6%, and 4.9% lifts in terms of R@1 of zero-shot text-to-video retrieval on MSR-VTT [79], DiDeMo [3], and MSVD [13] respectively. Besides, pretraining UMT [39] with our Panda-5M also outperforms the existing state-of-the-art methods [49, 50, 74] which are pretrained with much more vision-text data pairs (*i.e.*, >100M).

Table 6. **Zero-shot text-to-video generation.** We compare the zero-shot text-to-video generation of AnimateDiff [26] with the official weight (pretrained on 2.5 M videos) and our Panda-2M pretraining. We report FVD [68] on UCF101 [66] and CLIP similarity (CLIPSim) [76] on MSR-VTT [79]. We only compare with the models trained with less than 10M videos.

Method	(#) P-T Videos	UCF101	MSR-VTT
		FVD↓	CLIPSim↑
CogVideo [30]	5M	701.6	-
MagicVideo [92]	10M	699.0	-
LVDM [28]	18K	641.8	0.2751
ModelScope [70]	10M	639.9	**0.3000**
VideoLDM [8]	10M	550.6	-
AnimateDiff [26]	2.5M	499.3	0.2869
AnimateDiff [26]	**Panda2M (Ours)**	**421.9**	0.2880

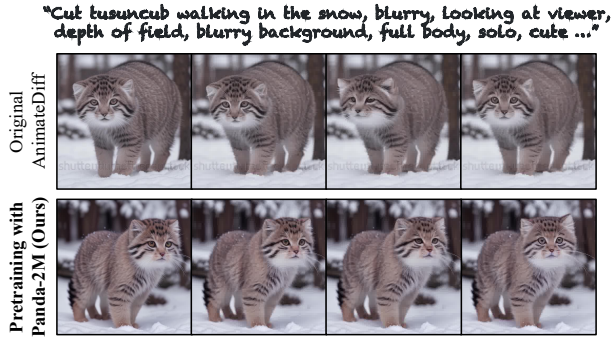

Figure 6. **Qualitative results of text-to-video generation.** We visualize the videos generated by the AnimateDiff [26] with official weight (top) and our Panda-2M pretraining (bottom). Note that the test prompt and the video sample of the original AnimateDiff (top) are directly from the project website of AnimateDiff.

4.3. Text-to-Video Generation

Experiment setup. To evaluate the effectiveness of text-to-video generation, we use AnimateDiff [26] as the base model and compare two weights: the officially released weight, which is trained on 2.5M text-video pairs, and the weight trained on our Panda-2M, a 2.5M subset of Panda-70M. We follow the official codebase and use Stable Diffusion v1.5 [59] (SD) as the base text-to-image (T2I) generator. During training, we fix T2I modules and only train the motion modeling modules. For each training video, we sample 16 frames with a stride of 4, and then resize and center-crop to 256×256px resolution.

Downstream datasets and evaluation metrics. To evaluate the models, we follow the evaluation protocols [8, 23, 65, 72] for zero-shot evaluation on UCF101 [66] and MSR-VTT [79]. Specifically, we generate 16-frame videos in 256×256px resolution. For UCF101 [66], we produce a text prompt for each class [23] and generate 10,000 videos which share the same class distribution as the original dataset [8, 72]. We compute Fréchet Video Distance (FVD) [68] on the I3D embeddings [84]. For MSR-VTT [79], we generate a video sample for each of the 59,800 test prompts [23, 65] and compute CLIP similarity (CLIPSim) [76]. We report the numbers in Table 6. We also show the generated video samples in Figure 6. To visualize the results, we follow the official codebase and replace SD T2I with personalized Dreambooth weight [60], TUSUN[3]. Note that the test prompt and the video sample from the AnimtateDiff with the official weight (top row in Figure 6) are directly from the project page of AnimateDiff.

Panda-2M pretraining consistently shows superior performance on both metrics compared to the official weight. As highlighted, our pretraining yields 77.4 lower FVD on UCF101 and outperforms state-of-the-art models pretrained on a dataset within a 10M scale in terms of FVD. Qualitatively, our pretraining weights can generate the video with a more meaningful motion and photorealistic appearance and do not include a watermark.

[3]https://civitai.com/models/33194/pallass-catmanul-lora

5. Conclusion and Limitations

This paper introduces Panda-70M, a large-scale video dataset with caption annotations. The dataset includes high-resolution and semantically coherent video samples. To caption 70M videos, we propose an automatic pipeline that can leverage multimodal information, such as video description, subtitles, and individual static video frames. We demonstrate that pretraining with Panda-70M can facilitate three downstream tasks: video captioning, video and text retrieval, and text-to-video generation.

Despite showing impressive results, the proposed dataset is still bound by a few limitations. First, we collect the videos from HD-VILA-100M [80], where most of the samples are vocal-intensive videos. Hence, the major categories of our dataset are news, television shows, documentary films, egocentric videos, and instructional and narrative videos. As our annotation pipeline does not require the presence of video subtitles, we list the collection of more unvocal videos as an important extension of this work.

Second, we focus on a fine-grained dataset where the video samples are semantically consistent so the caption can accurately express its semantics content without ambiguity. Nevertheless, it would limit the content diversity within a single video and also reduce average video duration, which might be hurtful to the downstream tasks, such as long video generation [9] and dense video captioning [71, 81]. Future efforts in building datasets with long videos and dense captions can benefit these downstream applications.

Risk mitigation. Prior to the release of the dataset, we used the internal automatic pipeline to filter out the video samples with harmful or violent language and texts that include drugs or hateful speech. We also use the NLTK framework to replace all people's names with "person".

References

[1] Pyscenedetect. https://github.com/Breakthrough/PySceneDetect. 4, 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. *NeurIPS*, 2022. 2, 3

[3] Lisa Anne Hendricks, Oliver Wang, Eli Shechtman, Josef Sivic, Trevor Darrell, and Bryan Russell. Localizing moments in video with natural language. In *ICCV*, 2017. 2, 3, 7

[4] Yogesh Balaji, Seungjun Nah, Xun Huang, Arash Vahdat, Jiaming Song, Karsten Kreis, Miika Aittala, Timo Aila, Samuli Laine, Bryan Catanzaro, et al. ediff-i: Text-to-image diffusion models with an ensemble of expert denoisers. *arXiv preprint*, 2022. 2, 3

[5] Satanjeev Banerjee and Alon Lavie. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *ACL workshop*, 2005. 6

[6] Hangbo Bao, Li Dong, Songhao Piao, and Furu Wei. BEit: BERT pre-training of image transformers. In *ICLR*, 2022. 3

[7] Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, et al. Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv preprint arXiv:2311.15127*, 2023. 3

[8] Andreas Blattmann, Robin Rombach, Huan Ling, Tim Dockhorn, Seung Wook Kim, Sanja Fidler, and Karsten Kreis. Align your latents: High-resolution video synthesis with latent diffusion models. In *CVPR*, 2023. 3, 8

[9] Tim Brooks, Janne Hellsten, Miika Aittala, Ting-Chun Wang, Timo Aila, Jaakko Lehtinen, Ming-Yu Liu, Alexei Efros, and Tero Karras. Generating long videos of dynamic scenes. *NeurIPS*, 2022. 8

[10] Minwoo Byeon, Beomhee Park, Haecheon Kim, Sungjun Lee, Woonhyuk Baek, and Saehoon Kim. Coyo-700m: Image-text pair dataset. https://github.com/kakaobrain/coyo-dataset, 2022. 2, 3

[11] Fabian Caba Heilbron, Victor Escorcia, Bernard Ghanem, and Juan Carlos Niebles. Activitynet: A large-scale video benchmark for human activity understanding. In *CVPR*, 2015. 2, 3

[12] Soravit Changpinyo, Piyush Sharma, Nan Ding, and Radu Soricut. Conceptual 12M: Pushing web-scale image-text pre-training to recognize long-tail visual concepts. In *CVPR*, 2021. 2

[13] David Chen and William B Dolan. Collecting highly parallel data for paraphrase evaluation. In *ACL*, 2011. 2, 3, 6, 7

[14] Jun Chen, Deyao Zhu, Xiaoqian Shen, Xiang Li, Zechu 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*, 2023. 3

[15] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, 2020. 5

[16] Tsai-Shien Chen, Wei-Chih Hung, Hung-Yu Tseng, Shao-Yi Chien, and Ming-Hsuan Yang. Incremental false negative detection for contrastive learning. In *ICLR*, 2022. 5

[17] Feng Cheng, Xizi Wang, Jie Lei, David Crandall, Mohit Bansal, and Gedas Bertasius. Vindlu: A recipe for effective video-and-language pretraining. In *CVPR*, 2023. 3, 7

[18] Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality, 2023. 6, 3

[19] Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. Scaling instruction-finetuned language models. *arXiv preprint*, 2022. 3

[20] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint*, 2018. 7, 5

[21] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021. 3, 7, 5

[22] Yuxin Fang, Wen Wang, Binhui Xie, Quan Sun, Ledell Wu, Xinggang Wang, Tiejun Huang, Xinlong Wang, and Yue Cao. Eva: Exploring the limits of masked visual representation learning at scale. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19358–19369, 2023. 6

[23] Songwei Ge, Seungjun Nah, Guilin Liu, Tyler Poon, Andrew Tao, Bryan Catanzaro, David Jacobs, Jia-Bin Huang, Ming-Yu Liu, and Yogesh Balaji. Preserve your own correlation: A noise prior for video diffusion models. In *ICCV*, 2023. 8

[24] Yuying Ge, Yixiao Ge, Xihui Liu, Dian Li, Ying Shan, Xiaohu Qie, and Ping Luo. Bridging video-text retrieval with multiple choice questions. In *CVPR*, 2022. 7

[25] Rohit Girdhar, Alaaeldin El-Nouby, Zhuang Liu, Mannat Singh, Kalyan Vasudev Alwala, Armand Joulin, and Ishan Misra. Imagebind: One embedding space to bind them all. In *CVPR*, 2023. 3, 4, 5, 1

[26] Yuwei Guo, Ceyuan Yang, Anyi Rao, Yaohui Wang, Yu Qiao, Dahua Lin, and Bo Dai. Animatediff: Animate your personalized text-to-image diffusion models without specific tuning. *arXiv preprint*, 2023. 8

[27] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *CVPR*, 2022. 3

[28] Yingqing He, Tianyu Yang, Yong Zhang, Ying Shan, and Qifeng Chen. Latent video diffusion models for high-fidelity long video generation. *arXiv preprint*, 2023. 8

[29] Jonathan Ho, William Chan, Chitwan Saharia, Jay Whang, Ruiqi Gao, Alexey Gritsenko, Diederik P Kingma, Ben Poole, Mohammad Norouzi, David J Fleet, et al. Imagen video: High definition video generation with diffusion models. *arXiv preprint*, 2022. 3

[30] Wenyi Hong, Ming Ding, Wendi Zheng, Xinghan Liu, and Jie Tang. Cogvideo: Large-scale pretraining for text-to-video generation via transformers. *arXiv preprint*, 2022. 8

[31] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *ICML*, 2021. 3

[32] Sangho Lee, Jiwan Chung, Youngjae Yu, Gunhee Kim, Thomas Breuel, Gal Chechik, and Yale Song. Acav100m: Automatic curation of large-scale datasets for audio-visual video representation learning. In *ICCV*, 2021. 2

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

[34] 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 *CVPR*, 2022. 7

[35] Junnan Li, Ramprasaath Selvaraju, Akhilesh Gotmare, Shafiq Joty, Caiming Xiong, and Steven Chu Hong Hoi. Align before fuse: Vision and language representation learning with momentum distillation. *NeurIPS*, 2021. 5

[36] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *ICML*, 2022. 2, 3

[37] 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*, 2023. 2, 3, 4, 5, 6

[38] Kunchang Li, Yinan He, Yi Wang, Yizhuo Li, Wenhai Wang, Ping Luo, Yali Wang, Limin Wang, and Yu Qiao. Videochat: Chat-centric video understanding. *arXiv preprint*, 2023. 2, 3, 4, 6

[39] Kunchang Li, Yali Wang, Yizhuo Li, Yi Wang, Yinan He, Limin Wang, and Yu Qiao. Unmasked teacher: Towards training-efficient video foundation models. *ICCV*, 2023. 2, 3, 4, 5, 7

[40] Chin-Yew Lin and Franz Josef Och. Automatic evaluation of machine translation quality using longest common subsequence and skip-bigram statistics. In *ACL*, 2004. 6

[41] Kevin Lin, Linjie Li, Chung-Ching Lin, Faisal Ahmed, Zhe Gan, Zicheng Liu, Yumao Lu, and Lijuan Wang. Swinbert: End-to-end transformers with sparse attention for video captioning. In *CVPR*, 2022. 6

[42] 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 *ECCV*, 2014. 6

[43] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *arXiv preprint*, 2023. 2, 3, 6

[44] Yang Liu, Samuel Albanie, Arsha Nagrani, and Andrew Zisserman. Use what you have: Video retrieval using representations from collaborative experts. *arXiv preprint*, 2019. 7

[45] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *ICCV*, 2021. 3

[46] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint*, 2017. 5, 7

[47] Ilya Loshchilov and Frank Hutter. SGDR: Stochastic gradient descent with warm restarts. In *ICLR*, 2017. 7

[48] 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*, 2020. 6

[49] 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 and captioning. *Neurocomputing*, 2022. 3, 7

[50] Yiwei Ma, Guohai Xu, Xiaoshuai Sun, Ming Yan, Ji Zhang, and Rongrong Ji. X-clip: End-to-end multi-grained contrastive learning for video-text retrieval. In *ACM MM*, 2022. 7

[51] Muhammad Maaz, Hanoona Rasheed, Salman Khan, and Fahad Shahbaz Khan. Video-chatgpt: Towards detailed video understanding via large vision and language models. *arXiv preprint*, 2023. 2, 3

[52] Antoine Miech, Dimitri Zhukov, Jean-Baptiste Alayrac, Makarand Tapaswi, Ivan Laptev, and Josef Sivic. Howto100m: Learning a text-video embedding by watching hundred million narrated video clips. In *ICCV*, 2019. 2, 3, 4

[53] OpenAI. Gpt-4 technical report. *arXiv preprint*, 2023. 2, 3

[54] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *ACL*, 2002. 6

[55] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, 2021. 3

[56] 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. *The Journal of Machine Learning Research*, 2020. 3

[57] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *ICML*, 2021. 2, 3

[58] Anna Rohrbach, Marcus Rohrbach, Niket Tandon, and Bernt Schiele. A dataset for movie description. In *CVPR*, 2015. 2, 3

[59] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022. 2, 3, 8

[60] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *CVPR*, 2023. 8

[61] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *NeurIPS*, 2022. 2, 3

[62] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. *NeurIPS*, 2022. 2, 3, 5

[63] Paul Hongsuck Seo, Arsha Nagrani, Anurag Arnab, and Cordelia Schmid. End-to-end generative pretraining for multimodal video captioning. In *CVPR*, 2022. 3

[64] Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *ACL*, 2018. 7

[65] Uriel Singer, Adam Polyak, Thomas Hayes, Xi Yin, Jie An, Songyang Zhang, Qiyuan Hu, Harry Yang, Oron Ashual, Oran Gafni, Devi Parikh, Sonal Gupta, and Yaniv Taigman. Make-a-video: Text-to-video generation without text-video data. In *ICLR*, 2023. 3, 8

[66] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*, 2012. 8

[67] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint*, 2023. 3

[68] Thomas Unterthiner, Sjoerd Van Steenkiste, Karol Kurach, Raphael Marinier, Marcin Michalski, and Sylvain Gelly. Towards accurate generative models of video: A new metric & challenges. *arXiv preprint*, 2018. 8

[69] Ramakrishna Vedantam, C Lawrence Zitnick, and Devi Parikh. Cider: Consensus-based image description evaluation. In *CVPR*, 2015. 6

[70] Jiuniu Wang, Hangjie Yuan, Dayou Chen, Yingya Zhang, Xiang Wang, and Shiwei Zhang. Modelscope text-to-video technical report. *arXiv preprint*, 2023. 8

[71] Teng Wang, Ruimao Zhang, Zhichao Lu, Feng Zheng, Ran Cheng, and Ping Luo. End-to-end dense video captioning with parallel decoding. In *ICCV*, 2021. 8

[72] Wenjing Wang, Huan Yang, Zixi Tuo, Huiguo He, Junchen Zhu, Jianlong Fu, and Jiaying Liu. Videofactory: Swap attention in spatiotemporal diffusions for text-to-video generation. *arXiv preprint*, 2023. 3, 8

[73] 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 *ICCV*, 2019. 2, 3

[74] Yi Wang, Kunchang Li, Yizhuo Li, Yinan He, Bingkun Huang, Zhiyu Zhao, Hongjie Zhang, Jilan Xu, Yi Liu, Zun Wang, et al. Internvideo: General video foundation models via generative and discriminative learning. *arXiv preprint*, 2022. 7

[75] Yi Wang, Yinan He, Yizhuo Li, Kunchang Li, Jiashuo Yu, Xin Ma, Xinhao Li, Guo Chen, Xinyuan Chen, Yaohui Wang, et al. Internvid: A large-scale video-text dataset for multimodal understanding and generation. *arXiv preprint arXiv:2307.06942*, 2023. 3

[76] Chenfei Wu, Lun Huang, Qianxi Zhang, Binyang Li, Lei Ji, Fan Yang, Guillermo Sapiro, and Nan Duan. Godiva: Generating open-domain videos from natural descriptions. *arXiv preprint*, 2021. 8

[77] Jialian Wu, Jianfeng Wang, Zhengyuan Yang, Zhe Gan, Zicheng Liu, Junsong Yuan, and Lijuan Wang. Grit: A generative region-to-text transformer for object understanding. *arXiv preprint*, 2022. 3

[78] Haiyang Xu, Qinghao Ye, Ming Yan, Yaya Shi, Jiabo Ye, Yuanhong Xu, Chenliang Li, Bin Bi, Qi Qian, Wei Wang, et al. mplug-2: A modularized multi-modal foundation model across text, image and video. *arXiv preprint*, 2023. 6, 7

[79] Jun Xu, Tao Mei, Ting Yao, and Yong Rui. Msr-vtt: A large video description dataset for bridging video and language. In *CVPR*, 2016. 2, 3, 6, 7, 8

[80] Hongwei Xue, Tiankai Hang, Yanhong Zeng, Yuchong Sun, Bei Liu, Huan Yang, Jianlong Fu, and Baining Guo. Advancing high-resolution video-language representation with large-scale video transcriptions. In *CVPR*, 2022. 1, 2, 3, 4, 8

[81] Antoine Yang, Arsha Nagrani, Paul Hongsuck Seo, Antoine Miech, Jordi Pont-Tuset, Ivan Laptev, Josef Sivic, and Cordelia Schmid. Vid2seq: Large-scale pretraining of a visual language model for dense video captioning. In *CVPR*, 2023. 3, 8

[82] Jiahui Yu, Zirui Wang, Vijay Vasudevan, Legg Yeung, Mojtaba Seyedhosseini, and Yonghui Wu. Coca: Contrastive captioners are image-text foundation models. *arXiv preprint*, 2022. 3, 5

[83] Jiahui Yu, Yuanzhong Xu, Jing Yu Koh, Thang Luong, Gunjan Baid, Zirui Wang, Vijay Vasudevan, Alexander Ku, Yinfei Yang, Burcu Karagol Ayan, Ben Hutchinson, Wei Han, Zarana Parekh, Xin Li, Han Zhang, Jason Baldridge, and Yonghui Wu. Scaling autoregressive models for content-rich text-to-image generation. *TMLR*, 2022. 2, 3

[84] Sihyun Yu, Jihoon Tack, Sangwoo Mo, Hyunsu Kim, Junho Kim, Jung-Woo Ha, and Jinwoo Shin. Generating videos with dynamics-aware implicit generative adversarial networks. *arXiv preprint arXiv:2202.10571*, 2022. 8

[85] Youngjae Yu, Jongseok Kim, and Gunhee Kim. A joint sequence fusion model for video question answering and retrieval. In *ECCV*, 2018. 7

[86] 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*, 2021. 3

[87] Rowan Zellers, Ximing Lu, Jack Hessel, Youngjae Yu, Jae Sung Park, Jize Cao, Ali Farhadi, and Yejin Choi. Merlot: Multimodal neural script knowledge models. *NeurIPS*, 2021. 2, 3

[88] Hang Zhang, Xin Li, and Lidong Bing. Video-llama: An instruction-tuned audio-visual language model for video understanding. *arXiv preprint*, 2023. 2, 3, 4, 5, 6

[89] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 2018. 4

[90] 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*, 2022. 5, 3

[91] Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. Bertscore: Evaluating text generation with bert. *arXiv preprint*, 2019. 6

[92] Daquan Zhou, Weimin Wang, Hanshu Yan, Weiwei Lv, Yizhe Zhu, and Jiashi Feng. Magicvideo: Efficient video generation with latent diffusion models. *arXiv preprint*, 2022. 3, 8

[93] Luowei Zhou, Chenliang Xu, and Jason Corso. Towards automatic learning of procedures from web instructional videos. In *AAAI*, 2018. 2, 3

[94] Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint*, 2023. 2, 3, 4, 6