Align and Prompt: Video-and-Language Pre-training with Entity Prompts

Dongxu Li[1,2], Junnan Li[1], Hongdong Li[2], Juan Carlos Niebles[1], Steven C.H. Hoi[1]
[1]Salesforce Research, [2]The Australian National University
dongxuli1005@gmail.com, {junnan.li,jniebles,shoi}@salesforce.com, hongdong.li@anu.edu.au

Abstract

Video-and-language pre-training has shown promising improvements on various downstream tasks. Most previous methods capture cross-modal interactions with a standard transformer-based multimodal encoder, not fully addressing the misalignment between unimodal video and text features. Besides, learning fine-grained visual-language alignment usually requires off-the-shelf object detectors to provide object information, which is bottlenecked by the detector's limited vocabulary and expensive computation cost.

*In this paper, we propose Align and Prompt: a new video-and-language pre-training framework (*ALPRO*), which operates on sparsely-sampled video frames and achieves more effective cross-modal alignment without explicit object detectors. First, we introduce a video-text contrastive (VTC) loss to align unimodal video-text features at the instance level, which eases the modeling of cross-modal interactions. Then, we propose a novel visually-grounded pre-training task, prompting entity modeling (PEM), which learns fine-grained alignment between visual region and text entity via an entity prompter module in a self-supervised way. Finally, we pretrain the video-and-language transformer models on large webly-source video-text pairs using the proposed VTC and PEM losses as well as two standard losses of masked language modeling (MLM) and video-text matching (VTM). The resulting pre-trained model achieves state-of-the-art performance on both text-video retrieval and videoQA, outperforming prior work by a substantial margin. Implementation and pre-trained models are available at* https://github.com/salesforce/ALPRO.

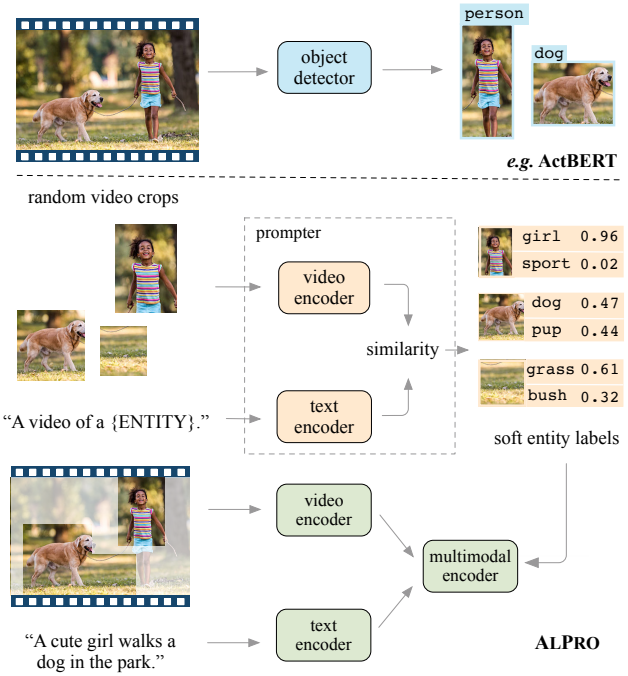

Figure 1. Generating supervision for region-entity alignment. **Above**: previous methods (*e.g.* ActBERT [56]) rely on object detectors with expensive computation cost and limited object categories, leaving text data unexploited. **Below**: ALPRO generates soft entity labels with a prompter module, which computes similarities between video crops and textual entity prompts. ALPRO requires no detector while taking advantage of video-text alignment to generate entity labels with a large vocabulary, thus strengthening the cross-modal learning.

1. Introduction

Video-and-language pre-training aims to jointly learn multimodal representations that transfer effectively to downstream tasks, such as *text-video retrieval* and *videoQA-video question answering*. Compared with images, videos usually contain more redundancy in consecutive frames. This challenges models on both capacity and computation efficiency. Most prior approaches [29, 34, 36, 38, 47, 56] circumvent the expensive computation overhead by using offline-extracted video features. Since the video feature extractors are fixed without finetuning, these approaches are suboptimal when transferring to distinct target domains. In contrast, recent emerging approaches [3, 25] sample frames sparsely from videos, which enable end-to-end pre-training and finetuning of video backbones. In this work, we adopt the sparse video-text pre-training paradigm considering their effectiveness on downstream tasks.

Despite their promising performance, current video-text pre-training models have several limitations. (1) The interaction between video and text features is commonly mod-

eled trivially using either dot-product [3,36,38,51] or cross-modal transformer encoders [25, 29, 47, 56]. However, features from individual modalities typically reside in different embedding spaces. Such misalignment makes it less effective to directly model cross-modal interaction. (2) Many visually-grounded pre-training tasks [29, 47] do not explicitly model fine-grained regional visual information (e.g. objects), which proves important for downstream tasks emphasizing on visual reasoning (e.g. videoQA). Although there are attempts which employ object detectors [7, 56] to generate pseudo-labels as supervision, they suffer from imprecise detections and a restricted number of object categories. For example, detectors trained on MSCOCO [30] recognize less than a hundred different categories. (3) The previous sparse pre-training model [25] is trained with image-text pairs using an image encoder, which makes it less effective in modeling temporal information.

In this paper, we tackle these challenges with a new video-and-language pre-training framework: Align and Prompt (ALPRO). Architecture-wise, ALPRO first encodes frames and text independently using a transformer-based video encoder and a text encoder, and then employs a multimodal encoder to capture cross-modal interaction. ALPRO learns both instance-level video-text alignment and fine-grained region-entity alignment. The instance-level alignment is learned by applying a video-text contrastive loss (VTC) on the unimodal features, which encourages paired video-text instances to have similar representations.

In order to better capture fine-grained visual information and strengthen region-entity alignment, ALPRO introduces a new visually-grounded pre-training task, called *prompting entity modeling*, where we ask the video-text model to predict entities appearing in randomly-selected video crops using jointly video and text inputs (see Figure 1). To address the unavailability of entity annotations, we design a standalone *entity prompter* module that generates reliable pseudo-labels. Specifically, the entity prompter consists of two unimodal encoders to extract video and text features, respectively. We first train the entity prompter using only VTC loss and freeze its parameters thereafter. Then during pre-training, we feed video crops and text prompts (*e.g.* "`A video of {Entity}.`") to the prompter, where each `Entity` is from the frequent nouns appearing in the pre-training corpus. We then compute the normalized similarity between the entity prompts and the video crop as the pseudo-label to supervise the pre-training.

Our key contributions are: **(1)** We introduce ALPRO, the first generic video-language pre-training method that learns effective cross-modal representations from *sparse* video frames and texts. **(2)** We introduce a video-text contrastive loss to better align instance-level unimodal representations, thus easing the modeling of cross-modal interaction. **(3)** We propose a novel visually-grounded pre-training task, prompting entity modeling, that enables the model to capture fine-grained region-entity alignment. **(4)** We demonstrate the effectiveness of ALPRO on both video-text retrieval and videoQA. ALPRO significantly improves over previous state-of-the-art methods, for example, achieving 3.0% and 5.4% absolute lift in recall scores on the fine-tuning and zero-shot text-video retrieval task on MSRVTT.

2. Related Work

Dense *versus* Sparse Video Representation. Consecutive frames in videos usually contain visually similar information. Such redundancy opens up a research question on how to learn effective video-and-language representations without excessive computation overhead. Most prior methods on text-video retrieval [13, 31, 42, 52, 54] and videoQA [11, 14, 24, 26] employ pre-trained visual backbones and extract video features for each frame *densely* yet offline. However, since visual backbones are usually pre-trained on image [22] and/or video datasets [20] without access to text, these features are less effective for video-and-language tasks. Besides, video feature extractors in these approaches are not finetuned on target task data, preventing features to easily adapt to different domains. In contrast, recent methods ClipBERT [25] and FiT [3] demonstrate more effective results by end-to-end finetuning the visual backbone with only a few *sparsely* sampled frames. However, ClipBERT is pre-trained with image-text data thus is less effective in aggregating information across frames, while FiT is a retrieval-specific architecture that does not naturally generalize to videoQA task. In this regard, our ALPRO is the first sparse pre-training architecture that tackles both tasks while in the meantime, demonstrating the benefit of pre-training on video-text pairs.

Video-and-language Pre-training. Apart from the canonical pre-training tasks, such as masked language modeling (MLM) [9, 25, 29, 34, 47, 56] and video-text matching (VTM) [29, 34], several methods [34, 36, 51] apply contrastive learning on *offline* extracted visual features. Without adapting the visual backbone, their ability to align cross-modal features remain limited. ALPRO jointly learns the unimodal and multimodal encoders, thus mitigating the disconnection in-between. In order to design effective visually-grounded pre-training tasks, VideoBERT [47] predicts centroids of vector quantizations of video features. Such unsupervised quantization is noisy per se while neglecting textual cues, which curtails its capabilities in learning cross-modal interactions. ActBERT [56] uses detectors to acquire object information. In addition to their computational inefficiency, detectors pre-trained on images usually have limited categories and compromised detection results on videos. In contrast, our proposed prompting entity modeling task is detector-free. By exploiting the instance-level video-text alignment, we can generate reliable entity

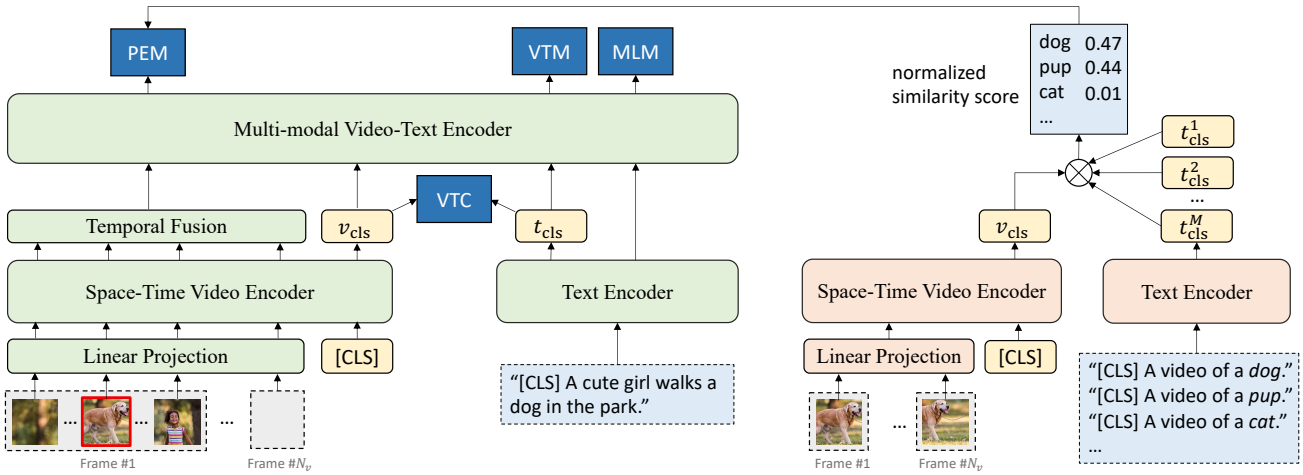

Figure 2. ALPRO pre-training framework. **Left**: the *video-language pre-training model* contains a space-time video encoder, a text encoder, and a multi-modal encoder, all of which are transformer-based. Besides two canonical objectives masked language modeling (MLM) and video-text matching (VTM), we introduce video-text contrastive loss (VTC) to learn instance-level video-text alignment, and prompting entity modeling (PEM) to learn fine-grained region-entity alignment. **Right**: the *prompter* which generates soft entity labels as supervision for PEM. The prompter consists of frozen unimodal encoders that are trained with VTC. During pre-training, it produces similarity scores between a randomly-selected video crop and a set of text prompts instantiated with entity names.

pseudo-labels with a large vocabulary, leading to more efficient and effective learning of region-entity alignment.

Zero-shot Visual Recognition with Prompts. There have been longstanding efforts to exploit text descriptions for learning visual recognition models. These include early efforts [4, 12, 23] that use text to learn attributes of images; [39, 40, 46] that map images to the pretrained text embedding space, and visual n-gram [27] that predicts text n-grams given image inputs. More recently, CLIP [43] instantiates prompt templates with label text of visual categories. It then predicts the category by computing the similarity between each image-prompt pairs. This inspires us to the design of entity prompter. Since the entity prompter is trained with the entire video-text corpus, during pre-training, it can provide additional entity information unavailable in the text description for each individual video-text pair, thus leading to better entity-informed video representations.

3. Video-Language Pre-training with ALPRO

In this section, we first introduce important constituent modules of ALPRO in Section 3.1. Then we present the pre-training objectives in Section 3.2, with a particular focus on the proposed video-text contrastive (VTC) loss and the prompting entity modeling (PEM) pre-training task. We introduce pre-training datasets in Section 3.3. Lastly, we describe important implementation details in Section 3.4.

3.1. ALPRO Architecture

Figure 2 gives an overview of ALPRO 's architecture. In particular, ALPRO consists of two main modules, a *video-language pre-training model* and a *prompter*. The prompter serves to generate soft entity labels to supervise the pre-training of the video-language model. Both modules contain their own video encoder and text encoder to extract features for video and text inputs, respectively. The pre-training model has an additional multimodal encoder to further capture the interaction between the two modalities. Details for each component are as follows.

Visual Encoder. We use a 12-layer TimeSformer$_{224}$ [5] to extract video features, with 224 the height and width of input frames. For N_v frames sparsely sampled from each input video, TimeSformer first partitions each frame into K non-overlapping patches, which are flattened and fed to a linear projection layer to produce a sequence of patch tokens. Learnable positional embeddings are also added to the patch tokens. Then the TimeSformer applies self-attention along the temporal and spatial dimensions separately in order, leading to per-frame features $\tilde{\boldsymbol{v}} \in \mathbb{R}^{N_v \times K \times d}$, with d the feature dimension. A temporal fusion layer (*i.e.* mean-pooling) is applied to $\tilde{\boldsymbol{v}}$ along the temporal dimension to aggregate per-frame features into video features. As the output of visual encoder, we obtain a sequence of visual embeddings: $\{\boldsymbol{v}_{\text{cls}}, \boldsymbol{v}_1, ..., \boldsymbol{v}_K\}$, with $\boldsymbol{v}_i \in \mathbb{R}^d$ and $\boldsymbol{v}_{\text{cls}}$ the embedding of the video `[CLS]` token.

Text Encoder. We use a 6-layer transformer [48] model to represent text tokens. Given an input text description of N_t tokens, the text encoder outputs an embedding sequence $\{\boldsymbol{t}_{\text{cls}}, \boldsymbol{t}_1, ..., \boldsymbol{t}_{N_t}\}$, with $\boldsymbol{t}_i \in \mathbb{R}^d$ and $\boldsymbol{t}_{\text{cls}}$ the embedding of the text `[CLS]` token. Similar to video encoder, we also add positional embeddings to the text tokens.

Multimodal Encoder. We employ a 6-layer transformer to model the interaction between video and text features

from the two unimodal encoders. Since positional embeddings are already injected in each unimodal encoder, we directly concatenate video and text features to feed the multimodal transformer. The outputs are multimodal embeddings $\{\boldsymbol{e}_{\text{cls}}, \boldsymbol{e}_1, ..., \boldsymbol{e}_{N_v+N_t}\}$, with $\boldsymbol{e}_i \in \mathbb{R}^d$. For notational convenience, we drop the multimodal embedding for the video `[CLS]` token as it is not used in pre-training losses.

3.2. Pre-training for ALPRO

We pre-train ALPRO with four objectives, including two canonical ones, *i.e*. masked language modeling (MLM) and video-text matching (VTM) as in [25,29,47,56]. In this section, we focus on presenting the new techniques in ALPRO, *i.e*. the video-text contrastive (VTC) loss and the prompting entity modeling (PEM) loss, while only briefly outlining MLM and VTM in Section 3.2.3, referring interested readers to [9, 25, 47] for details.

The motivation of both VTC and PEM is to strengthen cross-modal alignment between video and text. While VTC emphasizes on capturing instance-level alignment for video-text pairs, PEM encourages the model to align local video regions with textual entities. In the following, we introduce these two pre-training objectives in order.

3.2.1 Contrastive Video-Text Alignment

Existing sparse video-language pre-training models use either dot-product [3, 36, 38, 51] or rely entirely on a transformer encoder [25, 29, 47, 56] to model cross-modal interactions. However, since video and text features reside in different embedding spaces, such methods lead to less satisfactory alignment. To this end, we present a video-text contrastive (VTC) loss to align features from the unimodal encoders before sending them into the multimodal encoder. Specifically, given the embeddings of video and text `[CLS]` tokens, we optimize a similarity function between video V and text T:

$$s(V, T) = g_v(\boldsymbol{v}_{\text{cls}}) \cdot g_t(\boldsymbol{t}_{\text{cls}}), \tag{1}$$

such that paired video and text descriptions have higher similarity scores, where $g_v(\cdot)$ and $g_t(\cdot)$ are linear projections that transform the `[CLS]` embeddings to a common normalized low-dimensional (*e.g*. 256-d) space.

Following [17, 43], the contrastive loss considers matched pairs as positive and all others pairs that can be formed in a batch as negatives. For each input video-text pair $\langle V_i, T_i \rangle$, the video-text contrastive loss consists of two symmetric terms, one for video-to-text classification:

$$\mathcal{L}_{\text{v2t}} = -\log \frac{\exp(s(V_i, T_i)/\tau)}{\sum_{j=1}^{B} \exp(s(V_i, T_j)/\tau)} \tag{2}$$

and the other for text-to-video classification:

$$\mathcal{L}_{\text{t2v}} = -\log \frac{\exp(s(T_i, V_i)/\tau)}{\sum_{j=1}^{B} \exp(s(T_i, V_j)/\tau)}, \tag{3}$$

where τ is a learnable temperature parameter, and B is the batch size. The video-text contrastive loss is then defined as $\mathcal{L}_{\text{vtc}} = \frac{1}{2}(\mathcal{L}_{\text{v2t}} + \mathcal{L}_{\text{t2v}})$.

3.2.2 Prompting Entity Modeling

While masked language modeling has demonstrated its effectiveness on learning token-level text representations [9, 32], it remains a challenge to design its visually-grounded counterpart. As a result, the limited capabilities in visual reasoning adversely impact previous work on downstream tasks, especially those requiring region-level visual information such as objects. This is in particular an issue for existing video-language pre-training models [29, 36, 38, 38, 47], which usually retain only coarse-grained spatial information after pooling thus losing fine-grained visual cues. One exception is ActBERT [56] that attempts to use off-the-shelf object detectors to obtain regional features. Apart from its inefficiency, detectors trained with images tend to produce compromised detection results on video inputs. In addition, detectors are usually trained with restricted object categories (*e.g*. less than a hundred [30]), given its prohibitive expense to scale up the laborious annotations.

We introduce *prompting entity modeling* (PEM), a new visually-grounded pre-training task that improves the models' capabilities in capturing local regional information and strengthening cross-modal alignment between video regions and textual entities. Specifically, PEM requires a *prompter* module that generates soft pseudo-labels identifying entities that appear in random video crops. The pre-training model is then asked to predict the entity categories in the video crop, given the pseudo-label as the target.

The prompter serves to produce pseudo-labels of entity categories given a video crop, without dense annotations other than webly-sourced video-text pairs with possibly noisy alignment. To this end, we are inspired by CLIP [43] that learns image-text alignment from noisy pairs. Specifically, we first pre-train the prompter, which consists of two unimodal encoders, on video-text pairs with the VTC loss as in Section 3.2.1, and freeze its parameters thereafter.

The prompter maintains a predetermined list of M text prompts. Each text prompt is an instantiation of a template, *e.g*. "`A video of {ENTITY}`.", where `ENTITY` is a frequent noun in the pre-training corpus, such as `dog`, `grass`, `sky`, etc. After the prompter is pre-trained, it computes the `[CLS]` embedding for each text prompt as $\{\boldsymbol{t}^1_{\text{cls}}, \boldsymbol{t}^2_{\text{cls}}, ..., \boldsymbol{t}^M_{\text{cls}}\}$.

To generate entity labels, given one video input, we first obtain a random video crop $\hat{V}$ (*e.g*. the same spatial region across sampled frames) and its `[CLS]` embedding $\hat{\boldsymbol{v}}_{\text{cls}}$ from the prompter's video encoder. The prompter then computes an entity pseudo-label $\boldsymbol{q}_{\hat{V}} \in \mathbb{R}^M$ for the video crop as the softmax-normalized similarity between $\hat{\boldsymbol{v}}_{\text{cls}}$ and all the

prompt embeddings $\{\boldsymbol{t}_{\text{cls}}^m\}_{m=1}^M$:

$$q_{\hat{V},m} = \frac{\exp(s(\hat{V}, T_m)/\tau)}{\sum_{m=1}^M \exp(s(\hat{V}, T_m)/\tau)} \tag{4}$$

During pre-training of the video-language model, we apply mean pooling on the embeddings from the multimodal encoder that correspond to the spatial location of the video crop $\hat{V}$, denoted as $\boldsymbol{e}_{\hat{V}} \in \mathbb{R}^d$. We use a classifier (*e.g.* MLP) to compute the softmax-normalized entity prediction $\boldsymbol{p}_{\hat{V}}$. The prompting entity modeling loss is then defined as the cross-entropy between $\boldsymbol{p}_{\hat{V}}$ and $\boldsymbol{q}_{\hat{V}}$:

$$\mathcal{L}_{\text{pem}} = -\sum_{m=1}^{M} q_{\hat{V},m} \cdot \log p_{\hat{V},m} \tag{5}$$

Prompting entity modeling features a diverge range of entities while requiring no extra human annotations, which yields an efficient and scalable solution to generate visually-grounded regional supervisions for cross-modal learning.

3.2.3 Overall Pre-training Objectives

We also employ the widely-adopted masked language modeling (MLM) loss $\mathcal{L}_{\text{mlm}}$ and video-text matching $\mathcal{L}_{\text{vtm}}$ considering their effectiveness. The MLM objective utilizes both video and the contextual text to predict the masked text tokens. We randomly mask input tokens with a probability of 15% and replace them with a special token `[MASK]`. Video-text matching is a binary classification task which predicts whether a video and a text description are matched with each other. We use the multimodal `[CLS]` token $\boldsymbol{e}_{\text{cls}}$ as the joint representation of the video-text pair, and trains the model with a cross entropy loss. Negative samples are generated from non-parallel video-text pairs from the batch. Following [28], we employ contrastive hard negative mining to find more informative in-batch negatives for VTM. The overall pre-training objective of ALPRO is:

$$\mathcal{L} = \mathcal{L}_{\text{vtc}} + \mathcal{L}_{\text{pem}} + \mathcal{L}_{\text{mlm}} + \mathcal{L}_{\text{vtm}} \tag{6}$$

3.3. Pre-training Datasets

We pre-train our model with the webly-sourced dataset WebVid-2M [3], which contains 2.5M video-text pairs. In addition, as suggested by ClipBERT [25] and FiT [3], pre-training with image-pairs can improve spatial representations of videos, we therefore include CC-3M [45] into our pre-training corpus. During pre-training, we duplicate images from CC-3M to make static videos. This in total amounts to 5.5M video-text pairs, which is an order of magnitude less than the commonly-adopted HowTo100M [29, 36, 56] and of a comparable size to those used in [3, 25].

3.4. Implementation Details

We implement ALPRO in PyTorch [41]. In detail, we initialize both the spatial and temporal attention blocks of TimeSformer by reusing ViT-B/16 weights pre-trained on ImageNet-21k [10]. Text encoders are initialized using the first 6-layer of the $\text{BERT}_{\text{base}}$ model [9], and the multimodal encoder is initialized using the last 6-layers weights of $\text{BERT}_{\text{base}}$. We pre-train ALPRO for 100k iterations, roughly equivalent to 10 epochs, using a batch size of 256 on 16 NVIDIA A100 GPUs. We use AdamW [33] optimizer with a weight decay of 0.001. The learning rate is first warmed-up to $1e^{-4}$, then it follows a linear decay schedule. Since videos are usually of different aspect ratios, we first rescale them to 224×224. For each video, we sample 4 frames randomly as inputs to the visual encoder while preserving their orderings in-between. For PEM, we use POS tagger[1] and retain the top 1k most frequent nouns as the entity names. We obtain random video crops occupying $30\% - 50\%$ of the original spatial area as inputs to the prompter. We discard a pseudo-label if the most likely entity has a normalized similarity score smaller than 0.2.

4. Experiments

We evaluate the performance of ALPRO on text-video retrieval and video question answering tasks across four commonly-used datasets, introduced in Section 4.1. The purpose of the evaluation is three-fold. First, we demonstrate the effectiveness of major technical contributions (*i.e.* video-text contrastive loss and prompting entity modeling) in Section 4.2. We then compare the performance of ALPRO with previous methods, including task-specific and pre-training architectures, in Section 4.3 and Section 4.4, on retrieval and question answering tasks, respectively. Finally, we show ablation results on design choices and analyze model behaviors in Section 4.5.

4.1. Downstream Tasks and Datasets

Text-Video Retrieval. (i) **MSRVTT** [52] contains 10K videos with 200K text captions. We follow the common protocol [25, 29, 38, 54, 56] and use 7k videos for training and report results on the 1k test split [54]. (ii) **DiDeMo** [2] contains 10k videos from Flickr with 40k text descriptions. We follow [25, 31, 34] and evaluate paragraph-to-video retrieval, where sentence descriptions for each video are concatenated together as a single text query. We do not use the ground-truth proposals for temporal localization to ensure fair comparisons with previous work.

Video Question Answering. We focus on the task of open-ended video question answering. (i) **MSVD-QA** [50] is built upon videos and text descriptions from MSVD [6]. The MSVD-QA dataset has in total 1,970 videos and

[1] https://github.com/explosion/spaCy

Pre-training tasks	MSRVTT Retrieval				DiDeMo Retrieval				MSVD-QA	MSRVTT-QA
	R1↑	R5↑	R10↑	MdR↓	R1↑	R5↑	R10↑	MdR↓	Acc.↑	Acc.↑
w/o pre-training	16.5	42.8	57.9	7	9.5	29.1	42.5	14	41.5	39.6
MLM + VTM	28.5	53.0	66.8	5	29.8	57.7	69.7	4	43.3	40.9
MLM + VTM + PEM	30.3	56.7	67.8	4	31.0	61.8	73.5	3	**46.3**	41.8
MLM + VTM + VTC	32.8	59.2	70.3	3	**36.8**	64.7	77.4	**2**	45.5	41.9
MLM + VTM + PEM + VTC	**33.9**	**60.7**	**73.2**	**3**	35.9	**67.5**	**78.8**	3	45.9	**42.1**

Table 1. Evaluations of the proposed pre-training objectives on four downstream datasets. MLM: masked language modeling loss. VTM: video-text matching loss. PEM: prompting entity modeling loss. VTC: video-text contrastive loss. R@k denotes recall (%) with k retrieval efforts; MdR denotes median ranking for retrieved videos. We use acc. to denote accuracy.

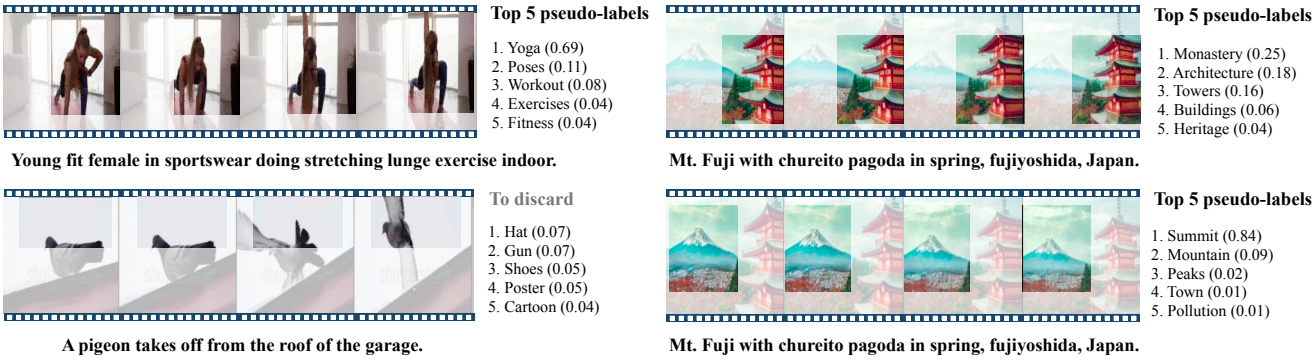

Figure 3. Examples of the pseudo-labels generated by the prompter (scores in bracket). The highlighted areas are fed to the prompter. Our method generates a diverse range of common entity categories that are not usually covered by object detectors, *e.g.* towers, summit, yoga. Besides, entity labels do not always appear in the text description, serving as a source of corpus-level supervision. **Bottom left**: a random crop that does not contain entities. The prompter thereby produces pseudo-labels with low similarities. During pre-training, we discard a pseudo-label if its most likely entity has a score less than 0.2. **Right**: labels generated for different crops from the same video.

50k question answer pairs, with 2,423 answer candidates. (ii) **MSRVTT-QA** [50] is built upon videos and captions from MSRVTT, which contains 10k videos with 243k open-ended questions and 1.5k answer candidates.

Finetuning Setups. On downstream tasks, ALPRO allows end-to-end finetuning of the video backbone with raw video frames as input. During finetuning, we randomly sample N frames per video, where $N = 8$ for retrieval and $N = 16$ for QA, with more ablations present in Section 4.4. Temporal position embeddings in TimeSformer are interpolated to accommodate different number of input frames. During inference, we sample frames uniformly to ensure reproducibility. To keep pre-training and finetuning setups consistent, we resize all the videos to 224×224 before feeding them into the model. Although this does not maintain the original aspect ratios, we observe no significant performance drop as our pre-training dataset contains videos of various aspect ratios. For finetuning on retrieval, we reuse the video-text matching head during pre-training and optimize the sum of both VTC and VTM losses. We obtain similarity scores from the output of VTM head during inference. For QA task, we add a simple MLP on the multimodal `[CLS]` token for classification and optmize the conventional cross-entropy loss between predictions and ground-truth answer labels. During inference, predictions are obtained as the answer with the highest probability. All the finetuning experiments are performed on 8 NVIDIA A100 GPUs, taking one to five hours to complete depending on the datasets. More training details can be found in the appendix.

4.2. Evaluation on the Proposed Methods

We first evaluate the impact of our main technical contributions (*i.e.* video-text contrastive loss and prompting entity modeling) in Table 1. Compared with pre-training using only MLM and VTM, both PEM and VTC substantially improve the performance across all the datasets. VTC is in particular useful for the retrieval task. The reason is that the VTC loss explicitly maximizes the instance-level similarity between positive video-text pairs, which is well-aligned with the goal of retrieval. We notice that PEM significantly improves the performance of videoQA, especially on MSVD-QA, due to its ability to learn finer-grained regional features. While enabling both PEM and VTC losses has complementary effects for most datasets, we also observe it leads to slightly worse accuracy on MSVD-QA. Our observation is that MSVD-QA contains more questions requiring region-level knowledge, including object categories (*e.g.* dough, swords), animal species (*e.g.* hare, eagle) and scenes (*e.g.* river, cliff), which can be well modeled using PEM, rendering the impact of VTC negligible. In contrast,

Method	PT datasets	R1↑	R5↑	R10↑	MdR↓
Finetuning					
JSFusion [54]	-	10.2	31.2	43.2	13
HT100M [38]	HT (100M)	14.9	40.2	52.8	9
ActBERT [56]	HT (100M)	16.3	42.8	56.9	10
NoiseEst. [1]	HT (100M)	17.4	41.6	53.6	8
HERO [29]	HT (100M)	16.8	43.4	57.7	-
ClipBERT [25]	COCO + VG (5.6M)	22.0	46.8	59.9	6
AVLNet [24]	HT (100M)	27.1	55.6	66.6	4
VideoClip [51]	HT (100M)	30.9	55.4	66.8	-
SupportSet [42]	HT (100M)	30.1	58.5	69.3	3
FiT [3]	Web2M + CC3M (**5.5M**)	31.0	59.5	70.5	3
ALPRO	Web2M + CC3M (**5.5M**)	**33.9**	**60.7**	**73.2**	**3**
Zero-shot					
HT100M [38]	HT (100M)	7.5	21.2	29.6	38
ActBERT [56]	HT (100M)	8.6	23.4	33.1	36
SupportSet [42]	HT (100M)	8.7	23.0	31.1	31
MIL-NCE [36]	HT (100M)	9.9	24.0	32.4	29.5
VideoClip [51]	HT (100M)	10.4	22.2	30.0	-
FiT [3]	Web2M + CC3M (**5.5M**)	18.7	39.5	51.6	10
ALPRO	Web2M + CC3M (**5.5M**)	**24.1**	**44.7**	**55.4**	**8**

Table 2. Comparisons with existing text-to-video retrieval methods with **finetuning** and **zero-shot** setups on **MSRVTT**. We follow the common partition with 7k training videos. *Methods using 9k training videos are greyed out*. Both partition protocols share the same 1k testing videos. R@k denotes recall (%) with k retrieval efforts; MdR denotes median ranking for retrieved videos. The pre-training datasets are HowTo100M (HT) [38], MS-COCO (COCO) [30], Visual Genome (VG) [21], WebVid2M (Web2M) [3] and Conceptual Captions (CC3M) [45].

MSRVTT-QA involves more coarse-grained visual information such as activities. As a result, using both PEM and VTC complements with each other on MSRVTT-QA.

Example Pseudo-labels. In Figure 3, we show examples of pseudo-labels generated by the prompter module. Our approach generates a more diverse range of entity categories beyond typical object classes from detection annotations. This is in particular beneficial when downstream tasks require a large vocabulary, such as open-ended videoQA.

4.3. Evaluation on Video-Text Retrieval

In Table 2 and Table 3, we compare ALPRO with existing methods using finetuning and zero-shot text-to-video retrieval on MSRVTT and DiDeMo datasets, respectively. ALPRO surpasses previous methods by a significant margin while exploiting orders of magnitude less video-text pairs with no human-written texts required. On both datasets, ALPRO obtains more than 6% lift in terms of R10 scores. Note that we do not compare with the work [35] which uses the powerful encoders from CLIP [43] pretrained on 400M *publicly inaccessible* image-text pairs. However, we remark that regardless of such weight reuse, Clip4Clip [35] has a similar architecture and objectives to FiT [3] while the latter uses an adapted TimeSformer as backbone. In this regard, ALPRO consistently surpasses FiT by a significant margin when pre-trained with the same amount of data shows its advantage over Clip4Clip.

Method	PT datasets	R1↑	R5↑	R10↑	MdR↓
Finetuning					
S2VT [49]	-	11.9	33.6	-	13
FSE [55]	-	13.9	36.0	-	11
CE [31]	-	16.1	41.1	-	8
MoEE [37]	-	16.1	41.2	55.2	8
ClipBERT [25]	COCO + VG (5.6M)	20.4	48.0	60.8	6
TT-CE [8]	-	21.6	48.6	62.9	6
FiT [3]	Web2M + CC3M (**5.5M**)	31.0	59.8	72.4	3
ALPRO	Web2M + CC3M (**5.5M**)	**35.9**	**67.5**	**78.8**	**3**
Zero-shot					
VideoCLIP [51]	HT (100M)	16.6	46.9	-	-
FiT [3]	Web2M + CC3M (**5.5M**)	21.1	46.0	56.2	7
ALPRO	Web2M + CC3M (**5.5M**)	**23.8**	**47.3**	**57.9**	**6**

Table 3. Comparisons with existing text-to-video retrieval methods with **finetuning** and **zero-shot** setups on **DiDeMo**. R@k denotes recall (%) with k retrieval efforts; MdR denotes median ranking for retrieved videos.

Method	PT datasets	MSRVTT	MSVD
E-SA [50]	-	29.3	27.6
ST-TP [16]	-	30.9	31.3
AMU [50]	-	32.5	32.0
Co-mem [14]	-	32.0	31.7
HME [11]	-	33.0	33.7
LAGCN [15]	-	-	34.3
HGA [19]	-	35.5	34.7
QUEST [18]	-	34.6	36.1
HCRN [24]	-	35.6	36.1
ClipBERT [25]	COCO + VG (5.6M)	37.4	-
SSML [1]	HT (100M)	35.1	35.1
CoMVT [44]	HT (100M)	39.5	42.6
VQA-T [53]	HTVQA (69M)	41.5	46.3
ALPRO	Web2M + CC3M (**5.5M**)	**42.1**	**45.9**

Table 4. Comparisons with existing methods on **MSRVTT-QA** and **MSVD-QA** in top-1 accuracy (%). VQA-T [53] uses 69M QA domain-specific data to pre-train their model while ALPRO uses an order of magnitude less video-text pairs from the web.

	MSRVTT-FT			MSRVTT-ZS			MSVD-QA
	R1↑	R10↑	MdR↓	R1↑	R10↑	MdR↓	Acc.↑
w/o ens.	32.7	73.1	3	22.6	52.3	9	45.0
with ens.	**33.9**	**73.2**	**3**	**24.1**	**55.4**	**8**	**45.9**

Table 5. Effect of ALPRO pre-training with and without prompt ensembling (ens.) on **MSVD-QA** and **MSRVTT text-video retrieval** with finetuning (FT) and zero-shot (ZS) setups.

#ent.	MSRVTT-FT			MSRVTT-ZS			MSVD-QA
	R1↑	R10↑	MdR↓	R1↑	R10↑	MdR↓	Acc.↑
∅	32.8	70.3	3	22.6	53.0	9	45.5
500	33.0	71.9	3	22.7	54.1	8	45.6
1000	33.9	73.2	3	24.1	55.4	8	45.9
2000	34.7	72.4	3	22.4	52.8	9	45.3

Table 6. Effect of the number of entities for PEM. We report results on **MSVD-QA and MSRVTT text-video retrieval** with finetuning (FT) and zero-shot (ZS) setups. The first row refers to the model trained with MLM+VTM+VTC (i.e w/o PEM).

4.4. Evaluation on Video Question Answering

Table 4 compares ALPRO with existing methods on open-ended video question answering datasets MSRVTT-QA and MSVD-QA. Most competitors have QA-specific architectures while that of ALPRO is generic for other video-language tasks, such as retrieval. We obtain on-par results with VQA-T [53], which exploits 69M *QA-specific* domain data for pre-training. In contrast, ALPRO uses only 5.5M video-text pairs from the web without domain knowledge. ALPRO surpasses other methods by a substantial margin, with 2.6% and 3.3% lift in accuracy. This demonstrates the competitive visual reasoning ability of ALPRO.

4.5. Ablations and Analysis

Prompt design and ensembling. Similar to [43], we observe that it is important to design and ensemble prompts with multiple templates. Without much engineering effort, we employ a preliminary set of prompt templates, such as "`A video of a {ENTITY}`", "`A footage of one {ENTITY}`" for video inputs; "`A photo of a {ENTITY}`" and "`A picture of the {ENTITY}`" for image inputs. In total, we design 12 templates for video and image inputs each. We build the ensemble by averaging over the $\boldsymbol{t}_{\text{cls}}$ embeddings of prompts instantiated with the same entity. The effect of prompt ensembling is shown in Table 5. Despite our minimal engineering efforts (we only experimented with a single set of templates), prompt ensembling demonstrates its importance in generating high-quality pseudo-labels. It is our future work to explore more prompt engineering strategies.

Effect of number of entities. We investigate the effect of the number of entities for PEM in Table 6. Compared with the model pre-trained with MLM+VTM+VTC, adding PEM brings consistent improvement with frequent entities.

#frms	MSRVTT-FT			MSRVTT-ZS			MSVD-QA
	R1↑	R10↑	MdR↓	R1↑	R10↑	MdR↓	Acc.↑
2	25.7	63.9	5	17.3	48.9	11	43.8
4	31.0	69.6	4	21.4	54.4	8	44.5
8	33.9	**73.2**	**3**	24.1	**55.4**	8	45.4
16	**34.2**	72.6	**3**	**24.7**	55.0	**7**	45.9

Table 7. Effect of the number of frames on **MSRVTT text-video retrieval and MSVD-QA**. More frames generally lead to better performance with 8-16 frames achieve a good trade-off between metrics and computation expense.

This suggests that the underlying principle of PEM to learn better region-entity alignment plays the essential role in its effectiveness. However, adding more low-frequency entities introduces noises in generating entity pseudo-labels, thus harming the pre-training.

Effect of number of frames. In Table 7, we show the results on downstream tasks with different numbers of input frames. Generally more frames lead to better performance, while such benefit saturates with more than 8 frames on the retrieval task. By sparsely sampling frames from the video and enabling end-to-end training of the visual backbone, ALPRO learns more effective representations than previous methods that use fixed offline features.

5. Conclusion

This paper proposes ALPRO , a new video-language pre-training framework that operates on sparsely-sampled video frames. ALPRO introduces video-text contrastive learning to align instance-level unimodal features, and prompting entity modeling for fine-grained region-entity alignment. We verify the efficiency and efficacy of ALPRO on multiple downstream datasets, where ALPRO achieves substantial performance improvement over existing models.

We believe ALPRO opens up a new direction for vision-language research, by exploiting the uptrending technique of prompting to generate semantic pseudo-labels. Here we list two potential ideas that can be further explored to improve ALPRO: (1) better prompt engineering / prompt tuning to improve the quality of entity pseudo-labels; (2) prompt-guided region selection with temporal information taken into consideration, which might improve the current way of random region selection. Last but not least, ALPRO is not restricted to the video domain, and can be naturally extended to image-text representation learning, or even image representation learning.

Limitations and Broader Impacts. There are still many limitations to ALPRO. While Section 5 discusses the technical aspect and proposes potential improvements, here we highlight the potential negative societal impact. Our pre-training data is collected from the web, which may contain unsuitable videos, harmful texts, and private information, which could leak into the pre-trained models. Additional model analysis is necessary before deploying it in practice.

References

[1] Elad Amrani, Rami Ben-Ari, Daniel Rotman, and Alex Bronstein. Noise estimation using density estimation for self-supervised multimodal learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 6644–6652, 2021. 7

[2] Lisa Anne Hendricks, Oliver Wang, Eli Shechtman, Josef Sivic, Trevor Darrell, and Bryan Russell. Localizing moments in video with natural language. In *Proceedings of the IEEE international conference on computer vision*, pages 5803–5812, 2017. 5

[3] Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman. Frozen in time: A joint video and image encoder for end-to-end retrieval. In *IEEE International Conference on Computer Vision*, 2021. 1, 2, 4, 5, 7

[4] Kobus Barnard, Pinar Duygulu, and David Forsyth. Clustering art. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, pages II–II. IEEE, 2001. 3

[5] Gedas Bertasius, Heng Wang, and Lorenzo Torresani. Is space-time attention all you need for video understanding? In *Proceedings of the International Conference on Machine Learning (ICML)*, July 2021. 3

[6] David Chen and William B 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, 2011. 5

[7] 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 *ECCV*, volume 12375, pages 104–120, 2020. 2

[8] Ioana Croitoru, Simion-Vlad Bogolin, Marius Leordeanu, Hailin Jin, Andrew Zisserman, Samuel Albanie, and Yang Liu. Teachtext: Crossmodal generalized distillation for text-video retrieval. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11583–11593, 2021. 7

[9] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. 2, 4, 5

[10] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2020. 5

[11] Chenyou Fan, Xiaofan Zhang, Shu Zhang, Wensheng Wang, Chi Zhang, and Heng Huang. Heterogeneous memory enhanced multimodal attention model for video question answering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1999–2007, 2019. 2, 7

[12] Ali Farhadi, Ian Endres, Derek Hoiem, and David Forsyth. Describing objects by their attributes. In *IEEE conference on computer vision and pattern recognition*, pages 1778–1785. IEEE, 2009. 3

[13] Valentin Gabeur, Chen Sun, Karteek Alahari, and Cordelia Schmid. Multi-modal transformer for video retrieval. In *Proceedings of the European Conference on Computer Vision*, pages 214–229. Springer, 2020. 2

[14] Jiyang Gao, Runzhou Ge, Kan Chen, and Ram Nevatia. Motion-appearance co-memory networks for video question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6576–6585, 2018. 2, 7

[15] Deng Huang, Peihao Chen, Runhao Zeng, Qing Du, Mingkui Tan, and Chuang Gan. Location-aware graph convolutional networks for video question answering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 11021–11028, 2020. 7

[16] Yunseok Jang, Yale Song, Youngjae Yu, Youngjin Kim, and Gunhee Kim. Tgif-qa: Toward spatio-temporal reasoning in visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2758–2766, 2017. 7

[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. 4

[18] Jianwen Jiang, Ziqiang Chen, Haojie Lin, Xibin Zhao, and Yue Gao. Divide and conquer: Question-guided spatio-temporal contextual attention for video question answering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 11101–11108, 2020. 7

[19] Pin Jiang and Yahong Han. Reasoning with heterogeneous graph alignment for video question answering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 11109–11116, 2020. 7

[20] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017. 2

[21] 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. 7

[22] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25:1097–1105, 2012. 2

[23] Christoph H Lampert, Hannes Nickisch, and Stefan Harmeling. Learning to detect unseen object classes by between-class attribute transfer. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 951–958. IEEE, 2009. 3

[24] Thao Minh Le, Vuong Le, Svetha Venkatesh, and Truyen Tran. Hierarchical conditional relation networks for video

question answering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9972–9981, 2020. 2, 7

[25] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7331–7341, 2021. 1, 2, 4, 5, 7

[26] Jie Lei, Licheng Yu, Mohit Bansal, and Tamara Berg. Tvqa: Localized, compositional video question answering. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 1369–1379, 2018. 2

[27] Ang Li, Allan Jabri, Armand Joulin, and Laurens van der Maaten. Learning visual n-grams from web data. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4183–4192, 2017. 3

[28] Junnan Li, Ramprasaath R. Selvaraju, Akhilesh Deepak Gotmare, Shafiq Joty, Caiming Xiong, and Steven Hoi. Align before fuse: Vision and language representation learning with momentum distillation. In *Advances in neural information processing systems*, 2021. 5

[29] 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, 4, 5, 7

[30] 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. 2, 4, 7

[31] Yang Liu, Samuel Albanie, Arsha Nagrani, and Andrew Zisserman. Use what you have: Video retrieval using representations from collaborative experts. In *British Machine Vision Conference*, 2019. 2, 5, 7

[32] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. 2019. 4

[33] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2018. 5

[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, 5

[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. 7

[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. 1, 2, 4, 5, 7

[37] Antoine Miech, Ivan Laptev, and Josef Sivic. Learning a text-video embedding from incomplete and heterogeneous data. *arXiv preprint arXiv:1804.02516*, 2018. 7

[38] 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, 4, 5, 7

[39] Mohammad Norouzi, Tomas Mikolov, Samy Bengio, Yoram Singer, Jonathon Shlens, Andrea Frome, Greg Corrado, and Jeffrey Dean. Zero-shot learning by convex combination of semantic embeddings. In *ICLR*, 2014. 3

[40] Mark Palatucci, Dean Pomerleau, Geoffrey E Hinton, and Tom M Mitchell. Zero-shot learning with semantic output codes. *Advances in Neural Information Processing Systems*, 22, 2009. 3

[41] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32:8026–8037, 2019. 5

[42] Mandela Patrick, Po-Yao Huang, Yuki Asano, Florian Metze, Alexander G Hauptmann, Joao F. Henriques, and Andrea Vedaldi. Support-set bottlenecks for video-text representation learning. In *International Conference on Learning Representations*, 2021. 2, 7

[43] 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. 3, 4, 7, 8

[44] Paul Hongsuck Seo, Arsha Nagrani, and Cordelia Schmid. Look before you speak: Visually contextualized utterances. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16877–16887, 2021. 7

[45] Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2556–2565, 2018. 5, 7

[46] Richard Socher, Milind Ganjoo, Christopher D Manning, and Andrew Ng. Zero-shot learning through cross-modal transfer. In *Advances in Neural Information Processing Systems*, pages 935–943, 2013. 3

[47] Chen Sun, Austin Myers, Carl Vondrick, Kevin Murphy, and Cordelia Schmid. Videobert: A joint model for video and language representation learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7464–7473, 2019. 1, 2, 4

[48] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia

Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017. 3

[49] Subhashini Venugopalan, Huijuan Xu, Jeff Donahue, Marcus Rohrbach, Raymond J Mooney, and Kate Saenko. Translating videos to natural language using deep recurrent neural networks. In *HLT-NAACL*, 2015. 7

[50] 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 *Proceedings of the ACM international conference on Multimedia*, pages 1645–1653, 2017. 5, 6, 7

[51] 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. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 6787–6800, 2021. 2, 4, 7

[52] Jun Xu, Tao Mei, Ting Yao, and Yong Rui. Msr-vtt: A large video description dataset for bridging video and language. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5288–5296, 2016. 2, 5

[53] Antoine Yang, Antoine Miech, Josef Sivic, Ivan Laptev, and Cordelia Schmid. Just ask: Learning to answer questions from millions of narrated videos. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1686–1697, 2021. 7, 8

[54] Youngjae Yu, Jongseok Kim, and Gunhee Kim. A joint sequence fusion model for video question answering and retrieval. In *Proceedings of the European Conference on Computer Vision*, pages 471–487, 2018. 2, 5, 7

[55] Bowen Zhang, Hexiang Hu, and Fei Sha. Cross-modal and hierarchical modeling of video and text. In *Proceedings of the European Conference on Computer Vision*, pages 374–390, 2018. 7

[56] Linchao Zhu and Yi Yang. Actbert: Learning global-local video-text representations. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8746–8755, 2020. 1, 2, 4, 5, 7