Learning text-to-video retrieval from image captioning

Lucas Ventura[1,2] · **Cordelia Schmid**[2] · **Gül Varol**[1]

Received: 3 April 2023

Abstract We describe a protocol to study text-to-video retrieval training with unlabeled videos, where we assume (i) no access to labels for any videos, i.e., no access to the set of ground-truth captions, but (ii) access to labeled images in the form of text. Using image expert models is a realistic scenario given that annotating images is cheaper therefore scalable, in contrast to expensive video labeling schemes. Recently, zero-shot image experts such as CLIP have established a new strong baseline for video understanding tasks. In this paper, we make use of this progress and instantiate the image experts from two types of models: a text-to-image retrieval model to provide an initial backbone, and image captioning models to provide supervision signal into unlabeled videos. We show that automatically labeling video frames with image captioning allows text-to-video retrieval training. This process adapts the features to the target domain at no manual annotation cost, consequently outperforming the strong zero-shot CLIP baseline. During training, we sample captions from multiple video frames that best match the visual content, and perform a temporal pooling over frame representations by scoring frames according to their relevance to each caption. We conduct extensive ablations to provide insights and demonstrate the effectiveness of this simple framework by outperforming the CLIP zero-shot baselines on text-to-video retrieval on three standard datasets, namely ActivityNet, MSR-VTT, and MSVD. Code and models will be made publicly available.

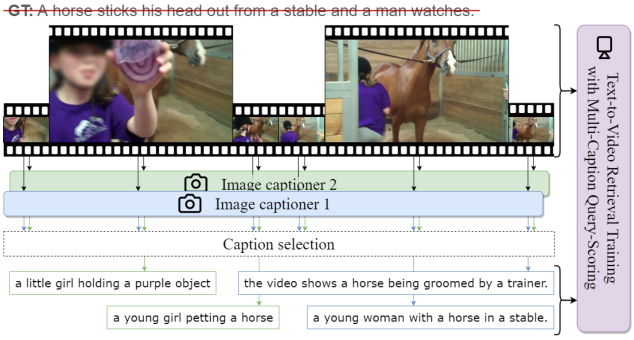

~~GT: A horse sticks his head out from a stable and a man watches.~~
Image captioner 2
Image captioner 1
Caption selection
a little girl holding a purple object
the video shows a horse being groomed by a trainer.
a young girl petting a horse
a young woman with a horse in a stable.
Text-to-Video Retrieval Training with Multi-Caption Query-Scoring

Fig. 1: **Framework:** Instead of using the ground-truth video caption, we extract *image* captions to automatically label *unlabeled video* frames, which we filter to obtain high-quality captions. The selected captions from multiple image captioners are incorporated into a text-to-video retrieval training where each video is paired with multiple caption labels.

1 Introduction

The research on automatic video understanding has witnessed a number of paradigm shifts recently. Following the rise of neural networks, the initial question was how to design an architecture to input spatio-temporal signals [49, 68]. Given the limited video training data, the focus then shifted to borrowing parameter initialization from image classification pretraining [7]. In an attempt to provide *video* pretraining, one line of work has made costly efforts to annotate video classification datasets [27]. On the other hand, the research community is moving away from closed-vocabulary recognition training as the progress in language modeling inspired advances in retrieval of visual data given open-vocabulary textual input, bridging the gap between

[1] LIGM, École des Ponts, Univ Gustave Eiffel, CNRS, France
[2] Inria, ENS, CNRS, PSL Research University, France
E-mail: lucas.ventura@enpc.fr
https://imagine.enpc.fr/~ventural/multicaps/

symbolic action categories and describing actions as text [26]. The latest shift was due to the huge scale of labeled image data, resulting in impressive zero-shot capability of image-text retrieval models on video action recognition tasks [55]. Now, the performance of CLIP-[55] or BLIP-initialized [32] image features (simply averaged over video frames) surpasses most previous works on a large number of video understanding tasks [32, 40, 73]. This makes researchers question and rethink where to put their efforts to improve video modeling. In this study, we focus on enhancing the *zero-shot* text-to-video retrieval performance of CLIP by making a realistic assumption that we have access to *image experts*, more specifically an image captioning model.

Fully-supervised methods for video retrieval are limited due to the high cost of video annotation. Even training with the web-scale video-text pairs [4] do not outperform CLIP image-text pretraining [8], despite the rich descriptions typed manually by humans with the motivation to sell their videos on stock websites. On the other hand, methods that learn from unlabeled videos often assume no access to *any* labels, even for images, with a particular focus on self-supervised training to use the structure of the data itself as the training signal [18, 22, 82]. In this paper, we ask the question of whether an external off-the-shelf image expert can provide the supervision signal. We explore the usability of recently released robust image captioners, namely ClipCap [46] and BLIP [32], which benefit from training with large-scale image-text pairs. For example, ClipCap uses both CLIP visual pretraining and GPT-2 language model pretraining [56]. When applied on video frames, we observe that, while noisy, the output texts contain high-quality descriptions, which motivates this exploration.

While the idea of using automatic image captions is appealing, incorporating such *noisy* labels for training introduces additional challenges. To address this issue, we first employ a filtering approach where we select the captions that better describe the frame by computing the CLIPScore metric [25]. Measuring such cross-modal similarity between the visual frame and the output text is similar in spirit to the filtering step in [32]. Furthermore, we ensemble multiple image captioners to obtain a larger pool of labels. We experimentally validate the benefit of these steps in our ablations.

In this work, we test whether off-the-shelf image captioning models can serve as an automatic labeling strategy for video retrieval tasks. We propose a simple framework to answer this question. Our main baseline, as well as our weight initialization, is CLIP [55]. We finetune this model such that video frame embeddings and the automatic captions map to the cross-modal joint space after contrastive retrieval training. Since one caption may not be representative of the video, we introduce multi-caption training to effectively use multiple textual labels per video, by extending the query-scoring method of [5]. This is to overcome the potential noise in automatic labels, as well as a way to augment data. Moreover, since our approach does not require manual labeling, we can go beyond a single dataset and combine multiple data sources during training. This particularly improves performance on smaller datasets. We demonstrate through experiments that our approach to pseudo-label unlabeled video frames with image captioning is a simple, yet effective strategy that boosts the performance over baselines.

Our contributions are three-fold: 1) We propose a new simple approach to train video retrieval models using automatic frame captions, which constitute free labels for supervision (see Figure 1). To the best of our knowledge, off-the-shelf captioning has not been used for such objectives in prior work at the time of conducting this research[1]. 2) We outperform the zero-shot state-of-the-art CLIP model on three text-to-video retrieval benchmarks. 3) We provide extensive ablations about the design choices on how to select high-quality captions, incorporating multiple image captioners, temporal pooling with multi-caption query-scoring, as well as combining multiple datasets. The code and models will be publicly available.

2 Related Work

We briefly overview relevant works on text-to-video retrieval, self-supervised learning on unlabeled videos, pseudo-labeling, and captioning.

Text-to-video retrieval. Methods for text-to-video retrieval only recently started to train end-to-end neural network models [4, 21] thanks to (i) the powerful initialization from ViT [16] and (ii) large-scale video datasets: noisy HowTo100M data [45] with ASR-based text supervision from speech, or more recently the cleaner manually annotated WebVid data [4]. The progress in text-to-image retrieval [12, 55] then triggered advances in text-to-video retrieval. Recent methods employ the CLIP [55] image backbone and explore the possibility of adding temporal modeling (e.g., CLIP2TV [20], CLIP4Clip [40], CLIP2Video [17], CLIP-ViP [79], TS2-Net [37], ViFi-CLIP [57]). Their results suggest that the simple averaging of embeddings over frames remains to be a strong baseline that is difficult to improve on. Several works have explored

[1] This paper is an extension of the preliminary work presented in [70].

fine-grained contrastive learning [84] for videos [41, 83], e.g., considering both frame-word and frame-sentence comparisons [41]. Bain et al. [5] presents a simple yet effective method to pool video frame representations with a weighted averaging based on query-scoring. In this work, we extend this method to use multiple captions instead of a single label per video. We also use CLIP [55] as our baseline, as well as our initialization. Similar to other retrieval methods [4, 40, 43], we employ a contrastive objective [51]. Unlike these approaches that assume manually annotated video data [4, 5, 40] or noisy speech signal [43, 77], we obtain our supervision from *automatic* captioning annotations. In our experiments, we show superior zero-shot performance over prior models trained on video-text pairs from HowTo100M [45] or WebVid [4].

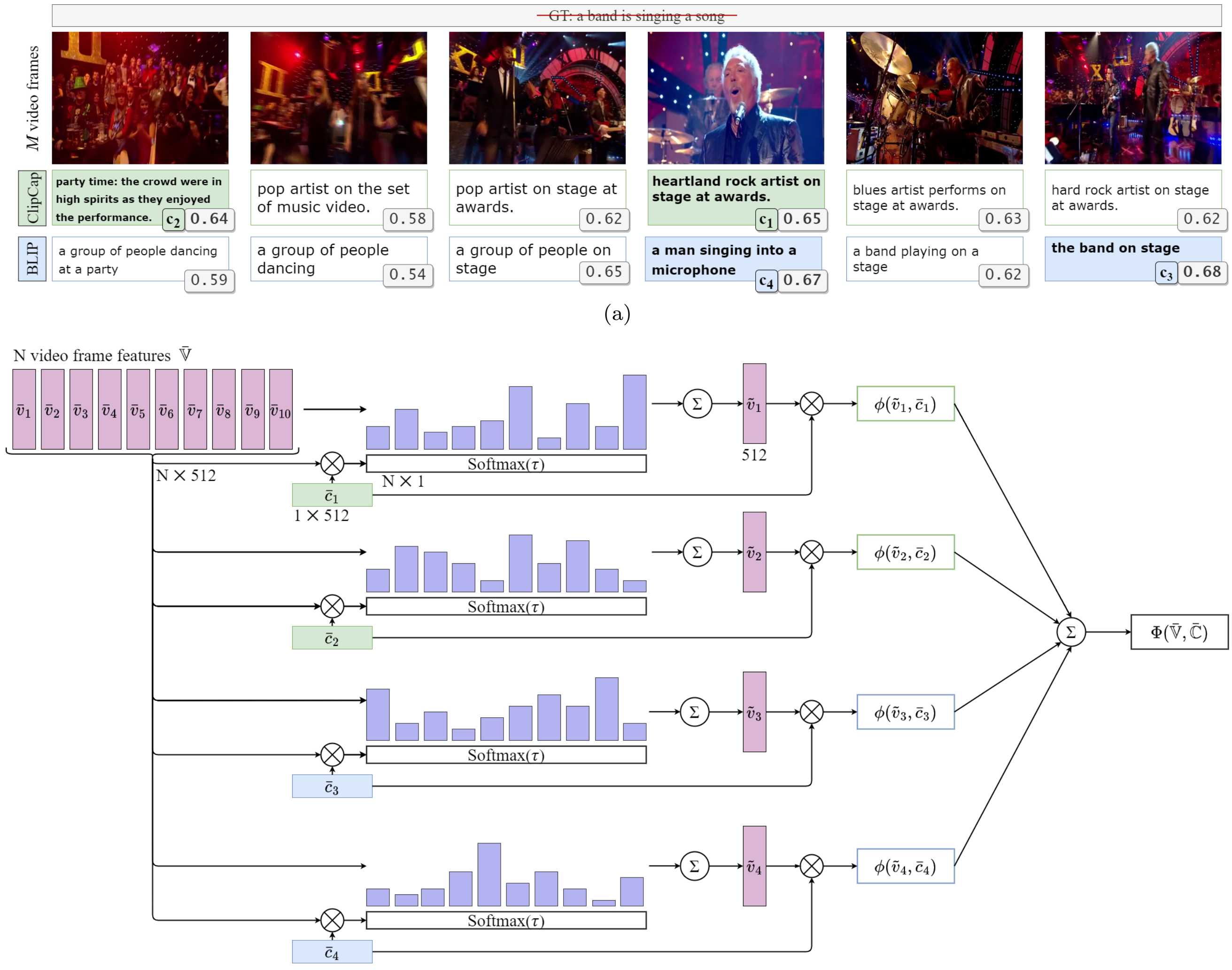
Self-supervised learning on unlabeled videos. A relevant line of work is representation learning on unlabeled videos, which is often referred to as self-supervised learning. In this category, several works [18, 22, 62, 66, 82] use instance discrimination for videos in a similar fashion with SimCLR [10] or BYOL [23] in the image setting. The majority of methods also make use of the multimodal nature of videos, e.g., incorporating the audio signal in the training [1, 2, 47, 54, 58]. A popular approach is to use the noisy speech signal in uncurated instructional videos such as HowTo100M [45]. The text obtained via ASR is directly considered as the corresponding label, which is then used within a contrastive objective [44, 53, 77]. [44] designs a multiple instance training, VideoCLIP [77] performs retrieval-augmented pretraining, and Support-set [53] defines a multi-task captioning objective. These self-supervised works may be complementary to our method, but our focus in this work is different in that we seek supervision from external image models that provide pseudo-labels, which can be considered as an alternative route to self supervision.

Pseudo-labeling. Our work is also relevant to pseudo-labeling (or self-labeling) approaches. Unlike the semi-supervised [30, 64, 65] or few-shot [76] setup considered in these works, our pseudo-labels do not require any annotations for the problem at hand. In particular, the concurrent work of [76] utilizes image experts to aid video-language learning, however, requiring a small set of labeled videos. In a similar fashion, VideoCC [48] exploits image-text datasets to assign automatic captions to videos for audiovisual retrieval, but is limited by the finite image captioning dataset source. Our work differs from [48] by *generating* captions for multiple video frames, rather than retrieving from such a finite set. While these two approaches may potentially be complementary, in our Appendix, we show that nearest neighbor retrieved captions perform worse than generated captions.

In text-image pretraining, BLIP [32] and BLIP-2 [31] employ a bootstrapping approach for image captioning, which falls into the semi-supervised category, i.e., they start training with a set of labeled images (whereas we never train on labeled videos). In fact, we employ BLIP as one of our image captioners to obtain automatic video labels. In our experiments, we also investigate the impact of using BLIP initialization as opposed to CLIP.

Captioning. There has been increasing interest in the task of generating text to describe a given visual content [3, 11, 13, 15, 39, 52, 61, 71, 80]. Although many works focus on integrating object information as additional guidance (e.g., Oscar [34], VLP [89]), such methods perform well on domains similar to that of the object detection model (e.g., COCO dataset [35]). ClipCap [46] shows robust performance across datasets of various domains without making use of an explicit object detection module. Instead, [46] makes use of two powerful pretrained models (CLIP [55] and GPT-2 [56]) and learns a mapping model between the image features and the language generation. More recently, BLIP [32], BLIP-2 [31] and CoCa [85] extend the contrastive CLIP training by jointly learning image captioning. Align and tell [75] also incorporates a video captioning head into their text-video retrieval model during training. OFA [74] further supports a variety of image-language tasks in a unified framework, where captioning can be performed by prompting the visual question answering model with 'What does the image describe?'. Very recently, CapDec [50] attaches a text decoder on top of the frozen CLIP image encoder by exploiting text-only data to train an autoencoder with the CLIP text encoder.

In our work, we employ ClipCap [46] and BLIP [32] as our image captioning experts, from which we obtain the supervision signal for unlabeled videos. While both of them are only image-based models, we find that their performance is satisfactory on video frames. The performance of video captioning models are currently behind those of image captioning approaches, mainly due to limited training data [52, 71]. Future work can explore them as their performance improves. Recent works of ClipVideoCap [81], Lavander [33], CLIP4Caption [67], HiREST [87], and TextKG [24] obtain promising results. However, our setup in this work considers no access to labeled videos.


~~GT: a band is singing a song~~
M video frames
ClipCap
BLIP
party time: the crowd were in high spirits as they enjoyed the performance.
c2 0.64
pop artist on the set of music video.
0.58
pop artist on stage at awards.
0.62
heartland rock artist on stage at awards.
c1 0.65
blues artist performs on stage at awards.
0.63
hard rock artist on stage at awards.
0.62
a group of people dancing at a party
0.59
a group of people dancing
0.54
a group of people on stage
0.65
a man singing into a microphone
c4 0.67
a band playing on a stage
0.62
the band on stage
c3 0.68
N video frame features $\bar{\mathbb{V}}$
$\bar{v}_1$ $\bar{v}_2$ $\bar{v}_3$ $\bar{v}_4$ $\bar{v}_5$ $\bar{v}_6$ $\bar{v}_7$ $\bar{v}_8$ $\bar{v}_9$ $\bar{v}_{10}$
N × 512
$\bar{c}_1$
1 × 512
N × 1
Softmax(τ)
Σ
$\tilde{v}_1$
512
$\phi(\tilde{v}_1, \bar{c}_1)$
$\bar{c}_2$
$\tilde{v}_2$
$\phi(\tilde{v}_2, \bar{c}_2)$
$\bar{c}_3$
$\tilde{v}_3$
$\phi(\tilde{v}_3, \bar{c}_3)$
$\bar{c}_4$
$\tilde{v}_4$
$\phi(\tilde{v}_4, \bar{c}_4)$
$\Phi(\bar{\mathbb{V}}, \bar{\mathbb{C}})$

(a)

(b)

Fig. 2: **Caption selection and multi-caption query-scoring (MCQS):** (a) To select the best captions for a given video, we first extract image captions from both ClipCap [46] and BLIP [32] models for M number of frames. We then compute the CLIPScore [5] (gray box), and finally select Top $K = 2$ captions for each captioner: c_1 and c_2 for ClipCap (highlighted in green), and c_3 and c_4 for BLIP (highlighted in blue). (b) MCQS takes a caption embedding $\bar{c}_l$ and weights the frame embeddings $\bar{v}_1 ... \bar{v}_N$ according to the query-scoring temporal poooling function f_p to obtain a video representation $\widetilde{v}_l$. Finally, we simply average the four similarities obtained with their respective query-scoring.

3 Training with automatic captions

In this section, we first describe how we obtain automatic captions for labeling videos, then present our multi-caption video retrieval training, and finally, give implementation details for our experimental setup.

The overview of our method is illustrated in Figure 2. In summary, we start by constructing a set of labels for each video, by applying image captioning models on video frames. Given these noisy frame-level captions (from multiple image captioners), we select the high-quality ones by sorting them according to their CLIPScore [25]. We adopt a contrastive video-text retrieval training using a multi-caption query-scoring approach, where we incorporate all the selected captions into the objective. Next, we detail these steps.

Selecting high-quality captions. Given an unlabeled training video v consisting of F frames, we select M frames from the video ($M \leq F$) and extract captions using I image captioners to form an initial set of labels $\mathbb{C} = \{\mathbb{C}_i\}_{i=1}^{I}$, where $\mathbb{C}_i = \{c_{i1}, c_{i2}, \ldots, c_{iM}\}$. We then obtain I textual descriptions per frame, resulting in a total of $M \times I$ labels per video.

While we investigate several variants of label formation from captions in our experiments, our final strategy is the following. We select a subset of the initial labels, mainly to eliminate noisy captions that do not well represent the corresponding video frame. To this end, we employ CLIPScore [25] as a way to measure the cross-modal similarity between a caption and its corresponding frame. For each captioner, we keep the top-K captions ($K < M$) with the highest CLIPScores, which gives us a remaining $L = K \times I$ labels per video. We refer to this subset as $\mathbb{C}'$. Note that some captions are repetitive across frames due to visual similarity within a video; we therefore conjecture that such a subset selection does not cause a significant loss in information.

Contrastive video retrieval objective with multi-caption query-scoring. In this work, we employ a relatively standard vision-language cross-modal training, where the goal is to find a joint space between videos and automatic captions. Given a video v, we compute visual embeddings $\bar{\mathbb{V}} = \{\bar{v}_n\}_{n=1}^{N}$ on N video frames ($N \leq F$) using a visual encoder $f_v : \bar{v}_n \rightarrow \mathbb{R}^d$. Similarly, we compute textual embeddings with the text encoder f_t from the corresponding set of labels $\mathbb{C}'$ to obtain positive text representations $\bar{\mathbb{C}} = \{\bar{c}_l\}_{l=1}^{L}$, where $\bar{c}_l \in \mathbb{R}^d$ (with the same embedding dimension as $\bar{v}_n$). To obtain a single video embedding, we perform temporal pooling over video frame representations. Inspired by the query-scoring introduced by [5], our pooling depends on the text representation, simply through weighted averaging, where frame weights are proportional to their similarity with the text. The pooled video embedding is then compared against the text to obtain a single similarity. Differently from [5], we have multiple texts $\bar{c}_l$. We therefore apply query-scoring multiple times, and obtain multiple similarities, which we combine by a simple mean operation (experiments with weighted mean do not yield improvements; see Section 4.2). More formally,

$$\varPhi(\bar{\mathbb{V}}, \bar{\mathbb{C}}) = \frac{1}{L} \sum_{l \in L} \phi(\tilde{v}_l, \bar{c}_l), \quad \text{where } \tilde{v}_l = f_p(\bar{\mathbb{V}}, \bar{c}_l), \tag{1}$$

represents a similarity between a set of video frame embeddings $\bar{\mathbb{V}}$ and a set of caption embeddings $\bar{\mathbb{C}}$, where $\phi(.)$ is the cosine similarity and f_p is the query-scoring [5] temporal pooling function also inputting the text:

$$f_p(\bar{\mathbb{V}}, \bar{c}_l) = \sum_{n \in N} w_n \bar{v}_n, \text{where} \quad w_n = \frac{e^{\phi(\bar{v}_n, \bar{c}_l)/\tau}}{\sum_{j \in N} e^{\phi(\bar{v}_j, \bar{c}_l)/\tau}}. \tag{2}$$

We set the softmax temperature hyperparameter $\tau = 0.1$ in our experiments.

From a batch of B visual-texts pair samples, $\{(\bar{\mathbb{V}}_1, \bar{\mathbb{C}}_1), (\bar{\mathbb{V}}_2, \bar{\mathbb{C}}_2), ..., (\bar{\mathbb{V}}_B, \bar{\mathbb{C}}_B)\}$, we train with a symmetric contrastive loss using InfoNCE [51], i.e., treating all other samples in the batch as negatives:

$$\mathcal{L}_{v2c} = -\frac{1}{B} \sum_{b \in B} \log \frac{\exp(\varPhi(\mathbb{V}_b, \mathbb{C}_b))}{\sum_{j \in B} \exp(\varPhi(\mathbb{V}_b, \mathbb{C}_j))} \tag{3}$$

$$\mathcal{L}_{c2v} = -\frac{1}{B} \sum_{b \in B} \log \frac{\exp(\varPhi(\mathbb{V}_b, \mathbb{C}_b))}{\sum_{j \in B} \exp(\varPhi(\mathbb{V}_j, \mathbb{C}_b))} \tag{4}$$

$$\mathcal{L} = \mathcal{L}_{c2v} + \mathcal{L}_{v2c}, \tag{5}$$

The final loss is the sum of video-to-captions ($\mathcal{L}_{v2c}$) and captions-to-video ($\mathcal{L}_{c2v}$) retrieval loss terms. Next, we detail the optimization procedure.

Implementation details. We instantiate two image captioners ($I = 2$) from ClipCap [46] and BLIP [32] models. ClipCap model is pretrained on the 3M images of the Google Conceptual Captions image-text dataset [63], using a MLP mapping between CLIP [55] image backbone and GPT-2 [56] text generation models. BLIP jointly trains for retrieval and captioning using 129M images (including a subset of LAION [60]) using a bootstrapping approach. We use the publicly available model, which is further finetuned on the COCO dataset [35]. Given one captioner, we extract $M = 10$ captions per video from equally spaced frames. We empirically set the number of high-quality captions to top $K = 2$ per captioner (i.e., $L = K \times I = 4$). On a single GTX1080 GPU, the captioning cost for ClipCap and BLIP is 0.65 fps and 0.93 fps, respectively.

We minimize the loss function in Eq. 5 using Adam [28] optimizer and a learning rate schedule with a cosine decay [38] as described in [40]. For ActivityNet, we train on 16 Tesla V100 GPUs for 10 epochs, with initial learning rate 10^{-5} and mini-batch size $B = 64$. For MSR-VTT and MSVD, we train on 4 NVIDIA GeForce GTX 1080 for 10 epochs, with initial learning rate 10^{-4} and mini-batch size $B = 16$.

The weights of our dual encoder model are initialized from CLIP [55] pretraining in all experiments unless explicitly stated otherwise, both for the image (f_v) and the text (f_t) encoders. The image encoder architecture follows ViT-B/16 [16] in all experiments. The text encoder architecture follows GPT-2 [56]. Both encoders are Transformer-based [69], operating with an embedding dimensionality of $d = 512$.

We resize the frames to 224 × 224 resolution before inputting them to the model. We use $N = 10$ random frame sampling during training based on segments as in [4, 72] (note that these do not necessarily match the $M = 10$ captions). The resulting spatio-temporal raw video input is of 224 × 224 × 10 dimensions. Each video frame is independently passed through the image encoder to obtain an embedding dimensionality of 512 using the output corresponding to the `[cls]` token. The temporal aggregation is obtained via query-scoring as explained above, i.e., weighted averaging over frames where the weights are obtained as frame-text similarity. The resulting video-level representation is therefore of dimensionality 512. During training, we use the multi-caption query-scoring method in Eq. 1. At test time, we compute the visual embeddings on the center spatial crop over 10 equally spaced frames. During evaluation, as we only have a single query text, multi-caption query scoring is not possible. We thus evaluate using the regular query-scoring method.

4 Experiments

We start with Section 4.1 by describing the datasets and evaluation metrics used to report the results of our experiments. We then present our ablations in Section 4.2, quantifying the effects of (i) the captioning model, (ii) caption selection, (iii) combining captioners, (iv) training with multiple captions per video, and (v) combining datasets. Next, we present a state-of-the-art comparison in Section 4.3, followed by experiments on BLIP initialization instead of CLIP in Section 4.4. Finally, we provide a qualitative analysis in Section 4.5, as well as a discussion on limitations in Section 4.6.

4.1 Datasets and evaluation metrics

We conduct experiments on three established benchmarks for text-to-video retrieval, namely ActivityNet [29], MSR-VTT [78], and MSVD [9] datasets.

ActivityNet Captions [29] contains 20k YouTube videos. Videos are segmented into 42k clips with an average length of 45s. We use the 10,009 videos from the training set, and evaluate on the "val1" split (4917 videos). Note that we extract equally spaced captions per clip, not per video.

MSR-VTT [78] is composed of 10k YouTube videos. The length of the videos varies from 10s to 32s, with an average of 15s. We train with the Training-9k split as in [4, 36, 40, 83], and report results on the 1k split with single video text-pairs as in [40, 86].

MSVD [9] consists of 1970 videos split into 1200 training, 100 validation, and 670 test videos. The dataset contains both short videos (∼1s) and long videos (∼60s). Given the small size of the dataset, we train using three different seeds and average the results on the test split.

	ActivityNet		MSR-VTT		MSVD	
	R@1	R@5	R@1	R@5	R@1	R@5
CLIP baseline [55]	23.4	49.3	32.8	55.7	39.4	64.6
Ours w/ OFA [74]	27.6	**55.6**	33.6	59.2	**41.1**	67.4
Ours w/ ClipCap [46]	26.7	53.5	34.7	59.8	40.6	68.9
Ours w/ BLIP [32]	**27.9**	54.2	**35.8**	**60.6**	**41.1**	**69.1**

Table 1: **Captioning models:** Training with automatic captions obtained with OFA [74], ClipCap [46], and BLIP [32] all improve over the zero-shot CLIP baseline [55] on all three text-to-video retrieval benchmarks. BLIP captions result in best performances.

As previously explained, even though these datasets contain ground-truth captions, we do *not* use them during training (see experiments in Section A on fully-supervised setting). We report the standard evaluation protocols: text-to-video (T2V) recall at rank 1 and 5 for all experiments. Recall at rank k (R@k) quantifies the number of times the correct video is among the top k results. Higher recall means better performance.

4.2 Ablation study

This work constitutes an exploratory study to test whether captions can provide a training signal for unlabeled videos. The answer is yes; however, there are certain design choices we make. Here, we provide ablations to measure the sensitivity to these decisions. More specifically, we investigate the effects of the captioning model and the quality of the captions provided to the model, To further improve the results, we make use of multiple captions per video during training, and combine datasets to train a single model.

(i) Captioning models. The first design choice is on the image captioning model to use. In Table 1, we present a comparative study experimenting with three recent captioning models: OFA [74], ClipCap [46] and BLIP [32]. More specifically, we use the best available model checkpoints: OFA-huge trained with 20M publicly available image-text pairs, ClipCap trained with Conceptual Captions, and BLIP-Large trained with 129M images, finetuned on COCO. Best results are obtained with BLIP, potentially due to the large amount of pretraining compared to the other two models. The results also demonstrate the effectiveness of using captions to improve over the strong CLIP baseline [55], where we average video frame embeddings using the frozen CLIP. Note that this is the same as the mean pooling method used in CLIP4Clip [40]. In this experiment, we randomly select one caption out of the two

Captioner	Caption selection	ActivityNet R@1	ActivityNet R@5	MSR-VTT R@1	MSR-VTT R@5	MSVD R@1	MSVD R@5
ClipCap	Rand(10)	25.1	51.9	31.8	55.2	39.8	68.5
	Middle 1	25.7	52.4	34.1	56.9	38.9	67.0
	Top 1	26.0	53.3	34.3	58.0	40.5	68.6
	Rand(Top 2)	**26.7**	**53.5**	**34.7**	**59.8**	**40.6**	**68.9**
	Rand(Top 3)	**26.7**	**53.5**	33.1	59.0	40.5	68.4
BLIP	Rand(10)	26.3	52.7	34.6	60.5	40.5	68.7
	Middle 1	25.7	52.4	33.2	57.8	40.1	**69.9**
	Top 1	27.6	**54.6**	34.9	60.3	**41.8**	68.3
	Rand(Top 2)	**27.9**	54.2	**35.8**	**60.6**	41.1	69.1
	Rand(Top 3)	27.8	54.2	35.6	59.5	40.9	68.2

Table 2: **Caption selection:** For both captioners, we compare training with a random caption at each epoch, training with only the middle frame caption, and training with different number of Top K captions (best CLIPScore [25]). Using CLIPScore filtering improves over using all the 10 captions or only using the middle one on both datasets. Selecting the Top 2 captions results in overall best performance.

	ActivityNet R@1	ActivityNet R@5	MSR-VTT R@1	MSR-VTT R@5	MSVD R@1	MSVD R@5
C	26.7	53.5	34.7	59.8	40.6	68.9
B	**27.9**	54.2	35.8	60.6	41.1	69.1
C+B	27.3	**54.5**	**36.5**	**61.5**	**41.7**	**70.0**

Table 3: **Combining two captioners:** We observe slight improvements when using captions from both ClipCap (C) and BLIP (B) over using them individually.

best captions during training. We next assess the influence of this selection.

(ii) Caption selection. Automatically generated captions vary in quality. We select captions with high image-text compatibility to eliminate potential noise in our training. The above image captioning models do not output a confidence score; therefore, we use CLIPScore [25] between the generated caption and the corresponding input video frame as a caption quality measure.

In Table 2, we evaluate whether such filtering is beneficial. In this experimental setup, we train with one caption as the video label. We experiment with five different variants per captioner: (a) randomly selecting one of the 10 extracted captions at each epoch, (b) using only the caption corresponding to the middle frame (i.e., same label in all epochs), (c) using only the best caption (i.e., top 1 based on the CLIPscore metric), (d) randomly selecting one of the 2 best captions at every epoch, (e) randomly selecting one of the 3 best captions at every epoch. The results support the idea that CLIPScore is an effective filtering method to keep the highest quality captions. On all three datasets, and on both captioners (ClipCap and BLIP), using the best caption(s) slightly improves over using all the captions or the middle one. Especially for ActivityNet, where the videos are relatively long, it is expected that the caption of the middle frame may not be representative of the video. However, there exists a trade-off between the number of captions and their quality. With more captions per video, we avoid overfitting as this may serve as data augmentation. On the other hand, the variance among the caption qualities starts to increase. We empirically find that taking the best two captions constitutes a good compromise, yielding a promising performance overall. However, the difference between top 1, 2, or 3 (last three rows) is not significant.

(iii) Combining captioners. One way to increase the number of captions per video without decreasing the quality of the captions is to use the best K captions from each captioner to form the label set. In Table 3, we test this hypothesis by taking two captioners ClipCap and BLIP, to then ensemble their labels. The results are slightly better than the performance of individual captioners on most metrics. One can potentially further extend to more captioners $I > 2$.

Note that we could also select the top K from all the captions combined from both captioners. This would be equivalent to taking the best 2 captions out of the 20 (10 per captioner). However, this leads to poorer results, perhaps due to the different CLIPScore distributions (slight preference for ClipCap potentially because of the CLIP backbone), and the tendency to output repetitive captions across frames for a given captioner. We provide further analysis in Section D.

(iv) Multi-caption query-scoring (MCQS). So far, we have only used one caption as a video label during each training iteration (even if this is randomly selected from a pool of 4). Here, we explore how to effectively combine multiple captions to get a richer video label, potentially capturing more global content beyond a single-frame caption. In Table 4, we compare multi-caption query-scoring (MCQS) with single-caption query-scoring (QS) for the 4 captions from ClipCap and BLIP as before.

We first evaluate the effect of QS for the uniform mean baselines (i.e., only at test time for the CLIP baseline, and also at training for one random caption baseline). Our first observation from Table 4 is that QS at evaluation marginally improves the baselines (33.9 vs 32.8 for CLIP, 37.6 vs 36.5 for Rand on MSR-VTT R@1). Training and evaluating with QS gives a further boost (38.3 vs 37.6).

In the last three rows of Table 4, we then explore three variants of our approach for using multiple cap-

Caption pooling	Temporal pooling		ActivityNet		MSR-VTT		MSVD	
	`train`	`eval`	R@1	R@5	R@1	R@5	R@1	R@5
CLIP baseline [55]		mean	23.4	49.3	32.8	55.7	39.4	64.6
		QS	23.8	50.0	33.9	57.3	38.5	64.6
Rand(4)	mean	mean	27.3	54.5	36.5	61.5	41.7	70.0
	mean	QS	27.8	55.0	37.6	64.3	41.9	70.0
	QS	QS	28.4	56.6	38.3	**64.8**	42.4	70.2
Concat(4)	QS	QS	**29.8**	**57.7**	27.3	50.9	35.1	62.6
Weighted(4)	MCQS	QS	29.0	57.0	38.6	63.2	41.5	**70.5**
Mean(4)	MCQS	QS	29.7	57.1	**39.0**	64.6	**42.5**	70.1

Table 4: **Multi-caption query-scoring:** Using all selected captions during training increases performance over only using one caption. The CLIP baseline and the model trained with randomly choosing one the 4 caption labels are evaluated with query-scoring (QS) for fair comparison. All models use Top-2 from both captioners (i.e., 4 captions in total from C+B).

Method	Data	Vision Backbone	ActivityNet		MSR-VTT		MSVD	
			R@1	R@5	R@1	R@5	R@1	R@5
CLIP w/ QS [55]	WiT	ViT-B/32	20.8	45.5	30.7	54.0	33.6	62.7
CLIP w/ QS [55]	WiT	ViT-B/16	23.8	50.0	33.9	57.3	38.5	64.6
ActBERT [90]	H	ResNet-101	-	-	8.6	23.4	-	-
SupportSet [53]	H	R(2+1)D-34	0.1	0.2	8.7	23.0	8.9	26.0
MIL-NCE [43]	H	I3D	-	-	9.9	24.0	-	-
VideoCLIP [77]	H	S3D	-	-	10.4	22.2	-	-
Frozen[4]	WebVid	ViT-B/16-time	-	-	24.7	46.9	-	-
CLIP4Clip [40]	WiT	ViT-B/32	-	-	31.2	53.7	-	-
VideoCC [48]	WiT+VCC	ViT-B/32	-	-	33.7	57.9	-	-
BLIP [32] (dual) †	B	ViT-B/16	26.3	52.5	35.7	59.2	35.2	63.3
BLIP [32] (cross-modal)	B	ViT-B/16	35.6	60.9	43.3	65.6	40.6	67.9
Ours (Self)	WiT+PL	ViT-B/16	29.7	57.0	39.0	64.6	42.5	70.0
Ours (Combined)	WiT+PL	ViT-B/16	**30.6**	**57.9**	**39.2**	**65.1**	**44.6**	**71.8**

Table 5: **Training on the combination of datasets:** We compare training and evaluating on the same dataset (Self), and training with the three combined datasets (Combined = ActivityNet + MSR-VTT + MSVD), and show that combining datasets removes the need of training three separate models and slightly improves the overall performance. We perform favorably compared to the state of the art on *zero-shot* retrieval (i.e., not using ground-truth video labels in downstream datasets). Colored lines are obtained from our implementation. † denotes results we obtained with the code from [32]. PL is short for pseudo-labels (using automatic captions). H: HowTo100M, VCC: VideoCC. B: COCO+VG+CC+SBU+LAION.

tions: a) concatenating captions into a single text and just using vanilla QS, b) weighted, or c) mean similarity pooling in MCQS. Simple concatenation significantly decreases the performance on MSR-VTT and MSVD, probably due to the distribution shift caused by the longer sentences during training (4 sentences during training vs 1 sentence at evaluation). On the other hand, ActivityNet results remain similar or even slightly improve as the standard evaluation protocol also concatenates ground-truth descriptions at test time [40]. The mean similarity pooling in MCQS obtains an overall improvement across datasets, over both CLIP and single-caption baselines. We observe a decrease in performance when dynamically weighting the similarities based on the ClipScore (with a softmax temperature of 0.1). We therefore keep the method simple and use the mean of similarities when jointly training with multiple captions in MCQS.

(v) Training with multiple datasets. Given that our framework does not require manually annotated videos, we are not constrained by the fixed size of a dataset's training split, and we can train with more data. In Table 5, we compare how the performance differs when: (i) training and evaluating on the same dataset (Self) versus (ii) training with more data by combining multiple datasets (Combined). The resulting combined training set has the following distribution in terms of number of video clips coming from each dataset: ∼79% ActivityNet, ∼19% MSR-VTT, and ∼2% from MSVD. The percentages represent the relative contribution of each dataset to the combined training set, derived from the total number of videos available in each dataset, with a uniform sampling approach that leads to a higher representation of ActivityNet due to its larger size. Such joint training improves performance moderately for the two relatively bigger datasets (ActivityNet and MSR-VTT), and more significantly for the small MSVD dataset. In the Appendix Section C.1, we also report cross-dataset evaluations (e.g., training with ActivityNet and evaluating on MSR-VTT). This experiment provides additional insights into the generalizability of our approach across different dataset domains. An additional advantage is to obtain a single model instead of multiple dataset-specific models. Future work can exploit including larger scale datasets provided sufficient computing resources.

4.3 Comparison with the state of the art

In Table 5, we summarize other zero-shot methods reporting performances mainly for MSR-VTT, and our method performs favorably against the state of the art. The rows that are colored are from our implementation, in comparable settings (e.g., using QS); uncolored rows correspond to other works. Red rows denote our baselines, green rows show our final models. Note that CLIP4Clip [40] zero-shot version is similar to our CLIP baseline [55] since they both use a frozen CLIP to mean-pool over frame embeddings. One difference is our use of query scoring, which was previously ablated in Table 4. Another difference may be due to different hyperparameters such as the number of frames ($N = 10$ in ours vs 12 in [40]). Note that in

Backbone (init.)	Cross-modal Encoder?	Method	ActivityNet R@1	ActivityNet R@5	MSR-VTT R@1	MSR-VTT R@5	MSVD R@1	MSVD R@5
CLIP [55]	No	Baseline	23.8	50.0	33.9	57.3	38.5	64.6
		Ours	29.7	57.1	39.0	64.6	**42.5**	**70.1**
BLIP [32] w/o COCO	No	Baseline	21.0	45.4	33.0	54.8	31.3	59.7
		Ours	23.4	48.0	33.8	60.5	33.7	62.2
	Yes	Baseline	32.1	57.1	41.4	63.3	39.6	63.9
		Ours	32.5	59.3	42.1	64.0	40.2	66.3
BLIP [32]	No	Baseline	28.2	56.3	37.4	62.2	37.7	67.3
		Ours	30.7	58.3	39.4	64.4	38.2	67.6
	Yes	Baseline	**35.1**	**60.6**	**43.5**	66.3	40.6	67.9
		Ours	34.6	**60.6**	**43.5**	**66.5**	42.2	68.5

Table 6: **Initialization with BLIP:** We show the comparison between the baseline versus our finetuning with automatic captions across various settings: CLIP/BLIP initialization, BLIP backbone with/without COCO finetuning, BLIP backbone with only the dual encoder or with subsequently reranking with its cross-modal encoder. Our method demonstrates improvements over the baseline across different initialization settings, but the gain is reduced as the baseline performance increases. For fairness, unlike the original BLIP evaluation, we use Query-Scoring (QS) when computing dual encoder similarities.

contrast to other works, we have access to the training videos (denoted with PL in Table 5), albeit without their corresponding ground-truth labels. On the other hand, some of the competitive methods require an external large source of videos such as WebVid [4] and VideoCC [48]. Others rely on noisy speech signal from the extensive HowTo100M data [44, 53, 77, 90], but their performances remain inferior.

Among prior works, BLIP [32] obtains higher performance than our method on MSR-VTT and ActivityNet. However, the BLIP model fundamentally differs from dual encoder approaches in that BLIP also contains a cross-modal encoder that is used for an additional image-text matching (ITM in their paper) as a classification task. The matching score from this classification head is then ensembled with the cosine similarity obtained by the dual encoder. Cross-modal encoders are known to perform better than dual encoders; however, they are less efficient [42]. We, therefore, gray out this line in Table 5 to highlight this difference. On the other hand, we compute the performance of the BLIP dual encoder, by considering only the cosine similarity between the unimodal embeddings (similar in spirit to CLIP). The result is much lower, for example for MSR-VTT 35.7 R@1, i.e., lower than both (i) their ensembled result 43.3 and (ii) our best model using only a dual encoder 39.2. We next extend our investigation to evaluate the applicability of our method on this more recent cross-modal BLIP encoder as an intialization instead of CLIP.

4.4 BLIP initialization

To evaluate the applicability of our method across various model initializations, we experiment with additional backbones beyond the primary CLIP model. In particular, we incorporate the BLIP model [32], which is available with and without COCO finetuning. The implementation details of BLIP, are summarized in Section E of the Appendix.

In Table 6, we compare (a) CLIP and BLIP, (b) two versions of BLIP pretraining, (c) both the efficient dual encoder version and the expensive reranking with the cross-modal version of BLIP as done in [32], (d) with/without our finetuning with automatic captions. Across all datasets and model configurations, we find that our finetuning with automatic captions consistently improves over the baselines, with the exception of the last two rows. The improvement is more significant for the CLIP backbone, than for BLIP where the baseline performance is already close to that of fully-supervised approaches (see Table A.1 of the Appendix). In other words, with greater baseline results of the underlying backbone, the more marginal the performance gains become.

We further note that the reranking operation with the cross-modal encoder, while generally leading to improved performance, is significantly less efficient than using the dual encoder alone. Specifically, in [32], an initial retrieval is obtained with the dual encoder, and the top-k ($k = 128$) retrieved videos are reranked with the costly cross-modal encoder. Without the cross-modal encoder, the CLIP-based model with our approach demonstrates superior performance (refer to rows with “No” under “Cross-modal Encoder” in Table 6). We also clarify that the BLIP baseline performances for both dual and cross-modal encoder configurations are slightly different when compared to Table 5, due to the incorporation of QS in the evaluation for a fair comparison; for example, MSR-VTT R@1 shows 37.4 vs 35.7 for the dual encoder and 43.5 vs 43.3 for the cross-modal encoder with and without QS, respectively. For the cross-modal encoder setup, QS is only used at the dual encoder retrieval stage, but not in reranking as the encoder inputs all frames without needing a temporal pooling as in [32].

We conclude the quantitative experiments by stating that pseudolabeling text-video retrieval datasets with image captioning allows finetuning text-to-image backbones with no manual annotation cost, which in turn substantially improves, for example over the frozen CLIP (e.g., 23.8 vs 30.6 on ActivityNet, 33.9 vs 39.2 on MSR-VTT, and 38.5 vs 44.6 on MSVD in Table 5).

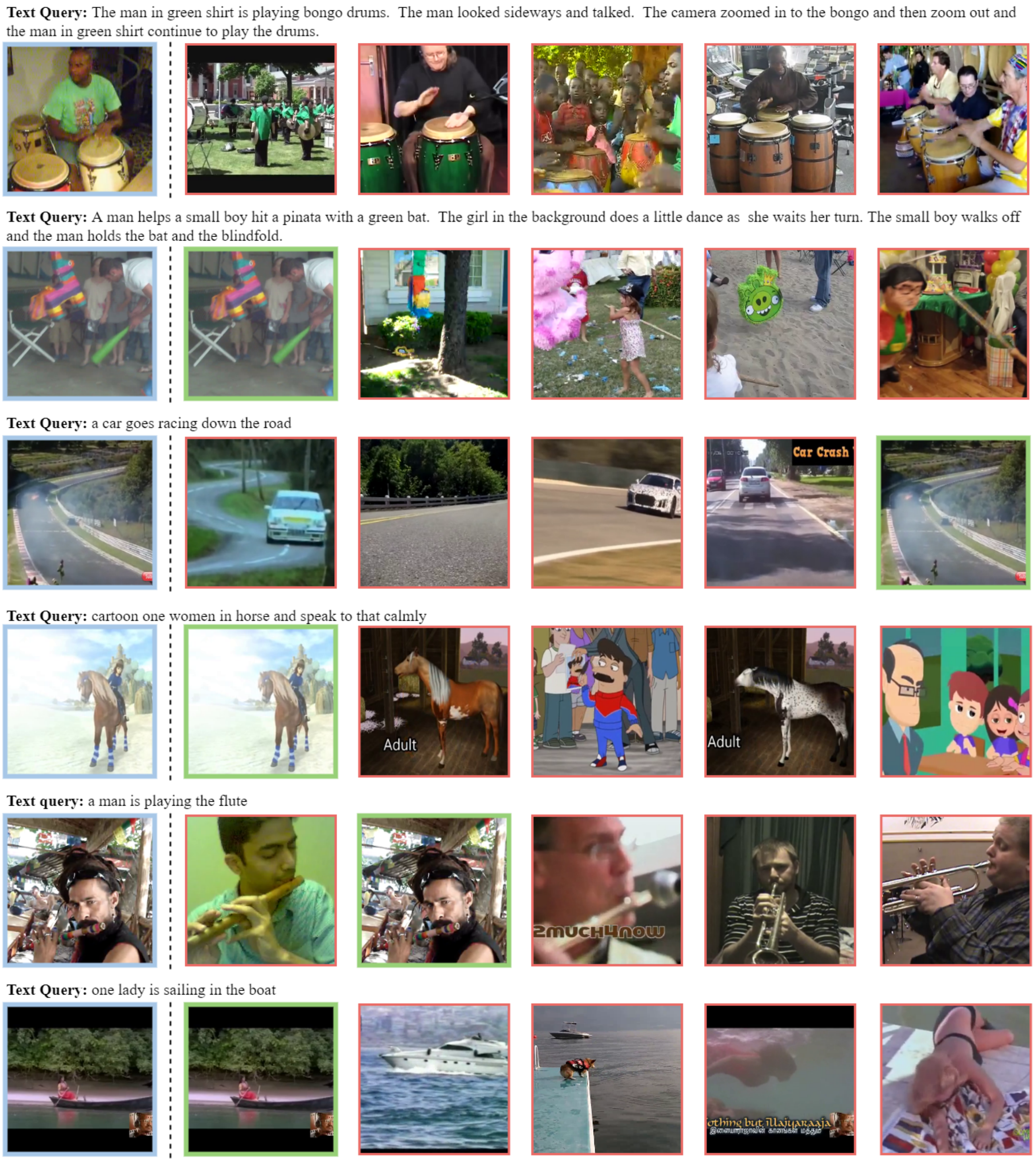

Text Query: The man in green shirt is playing bongo drums. The man looked sideways and talked. The camera zoomed in to the bongo and then zoom out and the man in green shirt continue to play the drums.
Text Query: A man helps a small boy hit a pinata with a green bat. The girl in the background does a little dance as she waits her turn. The small boy walks off and the man holds the bat and the blindfold.
Text Query: a car goes racing down the road
Car Crash
Text Query: cartoon one women in horse and speak to that calmly
Adult
Adult
Text query: a man is playing the flute
2MUCH4NOW
Text Query: one lady is sailing in the boat
othing but illaiyaraaja

Fig. 3: **Qualitative results:** We provide video retrieval results for our best model trained with the combination of the three datasets. The examples belong to the test sets of ActivityNet (first two rows), MSR-VTT (third and fourth rows), and MSVD (last two rows). For each example, we show the text query, the ground-truth video (first column, blue border), and top 5 retrieved videos from the gallery. Each video is only displayed using the middle frame, with a green border if matches the ground-truth video, or a red border otherwise. Overall, even cases where the correct video is not retrieved at the first rank, all the retrieved videos have similar semantic meaning with the text query.

4.5 Qualitative analysis

In Figure 3, we illustrate text-to-video results on several examples on all three datasets. For each test example, we display (a) the textual query, (b) the ground-truth video corresponding to the textual query (first column with blue border), (c) middle frames of the top 5 retrieved videos (in order from highest to lowest similarity), and (d) highlighted green border if the video matches the correct video, or a red border otherwise. Note that we only visualize the middle frame, which might not be representative for the overall video. We observe that most of the retrieved videos contain relevant information to the query text. For example, with the text query: "*cartoon one women in horse and speak to that calmly*", all the retrieved videos show cartoons. Moreover, sometimes even if the correct video is not ranked in the first position, there may be more than one valid option (e.g., the text query: "*a man is playing the flute*"). We provide more examples in Section F.

4.6 Limitations

Here, we discuss several limitations of this work. First, we note that image captioning does not necessarily capture the dynamic content of videos. In particular, some videos may only be recognized when observing several frames. Similarly, our temporal pooling approach remains simple, ignoring the order of frames. Temporal modeling efforts; however, do not yield gains for retrieval benchmarks [5]. As an attempt to incorporate temporal information, we performed preliminary analysis using text summarization techniques over the sequence of captions, but did not obtain consistent improvements (see Section B). Another limitation of our experiments is to train on the videos from the training set of a target dataset. Even if we do not use their labels, this setup ensures minimal domain gap. Future work can leverage large unlabeled video collections to remove this need.

5 Conclusion

We showed a simple yet effective framework to utilize an image captioning model as a source of supervision for text-to-video retrieval datasets. We demonstrated significant improvements over the strong zero-shot CLIP baseline with a comprehensive set of experiments. There are several promising directions for future development. One can explore the integration of more image experts beyond captioning, such as open-vocabulary object detection. The pseudolabeling approach could be extended to a wider variety of video data as mentioned in Section 4.6. The complementarity of self-supervised representation learning methods could be investigated to increase the supervision signal in unlabeled videos. Another future direction is to explore methods to combine the sequence of image captions into a single video caption.

References

1. Alayrac, J.B., Recasens, A., Schneider, R., Arandjelović, R., Ramapuram, J., De Fauw, J., Smaira, L., Dieleman, S., Zisserman, A.: Self-Supervised MultiModal Versatile Networks. In: NeurIPS (2020) 3
2. Alwassel, H., Mahajan, D., Korbar, B., Torresani, L., Ghanem, B., Tran, D.: Self-supervised learning by cross-modal audio-video clustering. In: NeurIPS (2020) 3
3. Anderson, P., He, X., Buehler, C., Teney, D., Johnson, M., Gould, S., Zhang, L.: Bottom-up and top-down attention for image captioning and visual question answering. In: CVPR (2018) 3
4. Bain, M., Nagrani, A., Varol, G., Zisserman, A.: Frozen in time: A joint video and image encoder for end-to-end retrieval. In: ICCV (2021) 2, 3, 5, 6, 8, 9
5. Bain, M., Nagrani, A., Varol, G., Zisserman, A.: A CLIP-hitchhiker's guide to long video retrieval. arXiv (2022) 2, 3, 4, 5, 11
6. Banerjee, S., Lavie, A.: METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In: Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization (2005) 17
7. Carreira, J., Zisserman, A.: Quo vadis, action recognition? A new model and the Kinetics dataset. In: CVPR (2017) 1
8. Castro, S., Heilbron, F.C.: FitCLIP: Refining large-scale pretrained image-text models for zero-shot video understanding tasks. arXiv (2022) 2
9. Chen, D., Dolan, W.: Collecting highly parallel data for paraphrase evaluation. In: Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (2011) 6, 19
10. Chen, T., Kornblith, S., Norouzi, M., Hinton, G.E.: A simple framework for contrastive learning of visual representations. In: ICML (2020) 3
11. Chen, X., Zitnick, C.L.: Learning a recurrent visual representation for image caption generation. arXiv:1411.5654 (2014) 3

12. Chen, Y.C., Li, L., Yu, L., Kholy, A.E., Ahmed, F., Gan, Z., Cheng, Y., Liu, J.: UNITER: Universal image-text representation learning. In: ECCV (2020) 2
13. Cho, J., Yoon, S., Kale, A., Dernoncourt, F., Bui, T., Bansal, M.: Fine-grained image captioning with CLIP reward. In: NAACL (2022) 3
14. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: Pre-training of deep bidirectional transformers for language understanding. In: NAACL (2019) 19
15. Donahue, J., Hendricks, L.A., Guadarrama, S., Rohrbach, M., Venugopalan, S., Darrell, T., Saenko, K.: Long-term recurrent convolutional networks for visual recognition and description. In: CVPR (2015) 3
16. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N.: An image is worth 16x16 words: Transformers for image recognition at scale. In: ICLR (2021) 2, 5
17. Fang, H., Xiong, P., Xu, L., Chen, Y.: CLIP2Video: Mastering video-text retrieval via image CLIP. arXiv:2106.11097 (2021) 2
18. Feichtenhofer, C., Fan, H., Xiong, B., Girshick, R.B., He, K.: A large-scale study on unsupervised spatiotemporal representation learning. In: CVPR (2021) 2, 3
19. Gabeur, V., Sun, C., Alahari, K., Schmid, C.: Multi-modal Transformer for Video Retrieval. In: ECCV (2020) 17
20. Gao, Z., Liu, J., Chen, S., Chang, D., Zhang, H., Yuan, J.: CLIP2TV: an empirical study on transformer-based methods for video-text retrieval. arXiv:2111.05610 (2021) 2
21. Ge, Y., Ge, Y., Liu, X., Li, D., Shan, Y., Qie, X., Luo, P.: Bridgeformer: Bridging video-text retrieval with multiple choice questions. In: CVPR (2022) 2
22. Gordon, D., Ehsani, K., Fox, D., Farhadi, A.: Watching the world go by: Representation learning from unlabeled videos. arXiv:2003.07990 (2020) 2, 3
23. Grill, J.B., Strub, F., Altch'e, F., Tallec, C., Richemond, P.H., Buchatskaya, E., Doersch, C., Pires, B.Á., Guo, Z.D., Azar, M.G., Piot, B., Kavukcuoglu, K., Munos, R., Valko, M.: Bootstrap your own latent: A new approach to self-supervised learning. In: NeurIPS (2020) 3
24. Gu, X., Chen, G., Wang, Y., Zhang, L., Luo, T., Wen, L.: Text with knowledge graph augmented transformer for video captioning. In: CVPR (2023) 3
25. Hessel, J., Holtzman, A., Forbes, M., Bras, R.L., Choi, Y.: CLIPScore: A reference-free evaluation metric for image captioning. In: EMNLP (2021) 2, 4, 5, 7, 16, 17
26. Ju, C., Han, T., Zheng, K., Zhang, Y., Xie, W.: Prompting visual-language models for efficient video understanding. arXiv:2112.04478 (2021) 2
27. Kay, W., Carreira, J., Simonyan, K., Zhang, B., Hillier, C., Vijayanarasimhan, S., Viola, F., Green, T., Back, T., Natsev, P., Suleyman, M., Zisserman, A.: The Kinetics human action video dataset. arXiv:1705.06950 (2017) 1
28. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. In: ICLR (2015) 5
29. Krishna, R., Hata, K., Ren, F., Fei-Fei, L., Niebles, J.C.: Dense-captioning events in videos. In: ICCV (2017) 6, 19
30. Lee, D.H.: Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In: ICMLW (2013) 3
31. Li, J., Li, D., Savarese, S., Hoi, S.: BLIP-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. arXiv:2301.12597 (2023) 3
32. Li, J., Li, D., Xiong, C., Hoi, S.C.H.: BLIP: bootstrapping language-image pre-training for unified vision-language understanding and generation. In: ICML (2022) 2, 3, 4, 5, 6, 8, 9, 15, 16
33. Li, L., Gan, Z., Lin, K., Lin, C.C., Liu, Z., Liu, C., Wang, L.: LAVENDER: Unifying video-language understanding as masked language modeling. arXiv (2022) 3
34. Li, X., Yin, X., Li, C., Zhang, P., Hu, X., Zhang, L., Wang, L., Hu, H., Dong, L., Wei, F., Choi, Y., Gao, J.: Oscar: Object-semantics aligned pre-training for vision-language tasks. In: ECCV (2020) 3
35. Lin, T.Y., Maire, M., Belongie, S., Bourdev, L., Girshick, R., Hays, J., Perona, P., Ramanan, D., Zitnick, C.L., Dollár, P.: Microsoft coco: Common objects in context. In: ECCV (2014) 3, 5
36. Liu, Y., Albanie, S., Nagrani, A., Zisserman, A.: Use what you have: Video retrieval using representations from collaborative experts. In: BMVC (2019) 6
37. Liu, Y., Xiong, P., Xu, L., Cao, S., Jin, Q.: TS2-Net: Token shift and selection transformer for text-video retrieval. In: ECCV (2022) 2
38. Loshchilov, I., Hutter, F.: Sgdr: Stochastic gradient descent with warm restarts. In: ICLR (2017) 5
39. Lu, J., Yang, J., Batra, D., Parikh, D.: Neural baby talk. In: CVPR (2018) 3
40. Luo, H., Ji, L., Zhong, M., Chen, Y., Lei, W., Duan, N., Li, T.: CLIP4Clip: An empirical study of CLIP

for end to end video clip retrieval. arXiv:2104.08860 (2021) 2, 3, 5, 6, 8, 17

41. Ma, Y., Xu, G., Sun, X., Yan, M., Zhang, J., Ji, R.: X-CLIP: End-to-end multi-grained contrastive learning for video-text retrieval. In: ACMMM (2022) 3
42. Miech, A., Alayrac, J.B., Laptev, I., Sivic, J., Zisserman, A.: Thinking fast and slow: Efficient text-to-visual retrieval with transformers. In: CVPR (2021) 9
43. Miech, A., Alayrac, J.B., Smaira, L., Laptev, I., Sivic, J., Zisserman, A.: End-to-end learning of visual representations from uncurated instructional videos. In: CVPR (2020) 3, 8
44. Miech, A., Alayrac, J.B., Smaira, L., Laptev, I., Sivic, J., Zisserman, A.: End-to-End Learning of Visual Representations from Uncurated Instructional Videos. In: CVPR (2020) 3, 9
45. Miech, A., Zhukov, D., Alayrac, J.B., Tapaswi, M., Laptev, I., Sivic, J.: HowTo100M: Learning a Text-Video Embedding by Watching Hundred Million Narrated Video Clips. In: ICCV (2019) 2, 3
46. Mokady, R., Hertz, A., Bermano, A.H.: ClipCap: CLIP prefix for image captioning. arXiv preprint arXiv:2111.09734 (2021) 2, 3, 4, 5, 6, 15, 16
47. Morgado, P., Vasconcelos, N., Misra, I.: Audio-visual instance discrimination with cross-modal agreement. arXiv:2004.12943 (2020) 3
48. Nagrani, A., Seo, P.H., Seybold, B.A., Hauth, A., Manen, S., Sun, C., Schmid, C.: Learning audio-video modalities from image captions. In: ECCV (2022) 3, 8, 9, 15
49. Ng, J.Y.H., Hausknecht, M., Vijayanarasimhan, S., Vinyals, O., Monga, R., Toderici, G.: Beyond short snippets: Deep networks for video classification. In: CVPR (2015) 1
50. Nukrai, D., Mokady, R., Globerson, A.: Text-only training for image captioning using noise-injected CLIP. arXiv:2211.00575 (2022) 3
51. van den Oord, A., Li, Y., Vinyals, O.: Representation learning with contrastive predictive coding. arXiv:1807.03748 (2018) 3, 5
52. Park, J.S., Rohrbach, M., Darrell, T., Rohrbach, A.: Adversarial inference for multi-sentence video description. In: CVPR (2019) 3
53. Patrick, M., Huang, P., Asano, Y.M., Metze, F., Hauptmann, A.G., Henriques, J.F., Vedaldi, A.: Support-set bottlenecks for video-text representation learning. In: ICLR (2021) 3, 8, 9
54. Piergiovanni, A.J., Angelova, A., Ryoo, M.S.: Evolving losses for unsupervised video representation learning. In: CVPR (2020) 3
55. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., Sutskever, I.: Learning transferable visual models from natural language supervision. In: ICML (2021) 2, 3, 5, 6, 8, 9, 15, 16, 17, 18
56. Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I.: Language models are unsupervised multitask learners. OpenAI blog (2019) 2, 3, 5
57. Rasheed, H., Khattak, M.U., Maaz, M., Khan, S., Khan, F.S.: Fine-tuned CLIP models are efficient video learners. In: CVPR (2023) 2
58. Recasens, A., Luc, P., Alayrac, J.B., Wang, L., Hemsley, R., Strub, F., Tallec, C., Malinowski, M., Patraucean, V., Altché, F., Valko, M., Grill, J.B., van den Oord, A., Zisserman, A.: Broaden your views for self-supervised video learning. arXiv:2103.16559 (2021) 3
59. Reimers, N., Gurevych, I.: Sentence-bert: Sentence embeddings using siamese bert-networks. In: EMNLP (2019) 15, 16
60. Schuhmann, C., Vencu, R., Beaumont, R., Kaczmarczyk, R., Mullis, C., Katta, A., Coombes, T., Jitsev, J., Komatsuzaki, A.: LAION-400M: open dataset of CLIP-filtered 400 million image-text pairs. In: Data Centric AI NeurIPS Workshop (2021) 5
61. Seo, P.H., Nagrani, A., Arnab, A., Schmid, C.: End-to-end generative pretraining for multimodal video captioning. In: CVPR (2022) 3
62. Sermanet, P., Lynch, C., Chebotar, Y., Hsu, J., Jang, E., Schaal, S., Levine, S.: Time-contrastive networks: Self-supervised learning from video. In: ICRA (2018) 3
63. Sharma, P., Ding, N., Goodman, S., Soricut, R.: Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In: ACL (2018) 5, 15
64. Singh, A., Chakraborty, O., Varshney, A., Panda, R., Feris, R., Saenko, K., Das, A.: Semi-supervised action recognition with temporal contrastive learning. In: CVPR (2021) 3
65. Sohn, K., Berthelot, D., Li, C.L., Zhang, Z., Carlini, N., Cubuk, E.D., Kurakin, A., Zhang, H., Raffel, C.: FixMatch: Simplifying semi-supervised learning with consistency and confidence. In: NeurIPS (2020) 3
66. Sun, C., Baradel, F., Murphy, K.P., Schmid, C.: Contrastive bidirectional transformer for temporal representation learning. arXiv:1906.05743 (2019) 3
67. Tang, M., Wang, Z., LIU, Z., Rao, F., Li, D., Li, X.: CLIP4Caption: CLIP for video caption. In:

ACMMM (2021) 3

68. Tran, D., Bourdev, L., Fergus, R., Torresani, L., Paluri, M.: Learning spatiotemporal features with 3D convolutional networks. In: ICCV (2015) 1
69. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L.u., Polosukhin, I.: Attention is all you need. In: NeurIPS (2017) 5
70. Ventura, L., Schmid, C., Varol, G.: Learning text-to-video retrieval from image captioning. In: CVPR Workshop on Learning with Limited Labelled Data for Image and Video Understanding (L3D-IVU) (2023) 2
71. Venugopalan, S., Rohrbach, M., Donahue, J., Mooney, R.J., Darrell, T., Saenko, K.: Sequence to sequence – video to text. In: ICCV (2015) 3
72. Wang, L., Xiong, Y., Wang, Z., Qiao, Y., Lin, D., Tang, X., Van Gool, L.: Temporal segment networks for action recognition in videos. IEEE Transactions on Pattern Analysis and Machine Intelligence **41**(11), 2740–2755 (2019) 5
73. Wang, M., Xing, J., Liu, Y.: ActionCLIP: A new paradigm for video action recognition. arXiv:2109.08472 (2021) 2
74. Wang, P., Yang, A., Men, R., Lin, J., Bai, S., Li, Z., Ma, J., Zhou, C., Zhou, J., Yang, H.: OFA: Unifying architectures, tasks, and modalities through a simple sequence-to-sequence learning framework. In: ICML (2022) 3, 6, 15
75. Wang, X., Zhu, L., Zheng, Z., Xu, M., Yang, Y.: Align and tell: Boosting text-video retrieval with local alignment and fine-grained supervision. IEEE Transactions on Multimedia (2022) 3
76. Wang, Z., Li, M., Xu, R., Zhou, L., Lei, J., Lin, X., Wang, S., Yang, Z., Zhu, C., Hoiem, D., Chang, S.F., Bansal, M., Ji, H.: Language models with image descriptors are strong few-shot video-language learners. arXiv (2022) 3
77. Xu, H., Ghosh, G., Huang, P.Y., Okhonko, D., Aghajanyan, A., Metze, F., Zettlemoyer, L., Feichtenhofer, C.: VideoCLIP: Contrastive pre-training for zero-shot video-text understanding. In: EMNLP (2021) 3, 8, 9
78. Xu, J., Mei, T., Yao, T., Rui, Y.: MSR-VTT: A large video description dataset for bridging video and language. In: CVPR (2016) 6, 19
79. Xue, H., Sun, Y., Liu, B., Fu, J., Song, R., Li, H., Luo, J.: CLIP-ViP: Adapting pre-trained image-text model to video-language representation alignment. arXiv (2022) 2
80. Yang, A., Nagrani, A., Seo, P.H., Miech, A., Pont-Tuset, J., Laptev, I., Sivic, J., Schmid, C.: Vid2seq: Large-scale pretraining of a visual language model for dense video captioning. In: CVPR (2023) 3
81. Yang, B., Zou, Y.: CLIP meets video captioners: Attribute-aware representation learning promotes accurate captioning. arXiv:2111.15162 (2021) 3
82. Yang, C., Xu, Y., Dai, B., Zhou, B.: Video representation learning with visual tempo consistency. arXiv:2006.15489 (2020) 2, 3
83. Yang, J., Bisk, Y., Gao, J.: TACo: Token-aware cascade contrastive learning for video-text alignment. arXiv (2021) 3, 6
84. Yao, L., Huang, R., Hou, L., Lu, G., Niu, M., Xu, H., Liang, X., Li, Z., Jiang, X., Xu, C.: FILIP: Fine-grained interactive language-image pre-training. In: ICLR (2022) 3
85. Yu, J., Wang, Z., Vasudevan, V., Yeung, L., Seyedhosseini, M., Wu, Y.: CoCa: Contrastive captioners are image-text foundation models. Transactions on Machine Learning Research (2022) 3
86. Yu, Y., Kim, J., Kim, G.: A joint sequence fusion model for video question answering and retrieval. In: ECCV (2018) 6
87. Zala, A., Cho, J., Kottur, S., Chen, X., Oğuz, B., Mehdad, Y., Bansal, M.: Hierarchical video-moment retrieval and step-captioning. In: CVPR (2023) 3
88. Zhang, B., Hu, H., Sha, F.: Cross-modal and hierarchical modeling of video and text. In: ECCV (2018) 17
89. Zhou, L., Palangi, H., Zhang, L., Hu, H., Corso, J.J., Gao, J.: Unified vision-language pre-training for image captioning and VQA. In: AAAI (2020) 3
90. Zhu, L., Yang, Y.: ActBERT: Learning global-local video-text representations. In: CVPR (2020) 8, 9

APPENDIX

This appendix provides experiments with the fully-supervised setting (Section A), results with alternative methods (Section B), additional evaluations (Section C), analyses on selecting captions and combining captioners (Section D), implementation details about the BLIP initialization experiment (Section E), additional qualitative results (Section F), and a data availability statement (Section G).

A Fully-supervised setting

While our focus is on the zero-shot setting, where labeled video data is not available, it is worth noting that for small-scale datasets, annotation costs may not be prohibitively high allowing for fully-supervised settings. In the following, we report experiments by training with the ground-truth captions in the datasets we use,

Method	Data	Vision backbone	ActivityNet		MSR-VTT		MSVD	
			R@1	R@5	R@1	R@5	R@1	R@5
CLIP	WiT	ViT-B/16	23.8	50.0	33.9	57.3	38.5	64.6
Ours (Self)	WiT+PL	ViT-B/16	29.7	57.1	39.0	64.6	42.5	70.1
Ours (Comb.)	WiT+PL	ViT-B/16	30.6	57.9	39.2	65.1	44.6	71.8
GT	WiT+GT	ViT-B/16	36.4	66.5	42.9	70.9	43.4	74.3
GT w/ QS	WiT+GT	ViT-B/16	35.1	64.9	44.0	70.5	46.0	73.9
GT w/ QS	WiT+GT+PL	ViT-B/16	**38.3**	**68.8**	**45.4**	**72.4**	**47.0**	**75.0**

Table A.1: **Fully-supervised setting:** Comparison of Baseline, Ours, and training with Ground Truth (GT) captions. PL denotes training with the dataset videos without ground truth labels. The last row shows the results obtained by fine-tuning the *Ours (Comb.)* model from Table 5.

Caption pooling	Temporal pooling		MSR-VTT	
	train	eval	R@1	R@5
CLIP [55]	-	QS	23.8	50.0
Random(GT)	mean	QS	42.9	70.9
Mean(GT)	MCQS	QS	**44.9**	**73.3**

Table A.2: **Multi-caption query-scoring training on MSR-VTT:** Comparison of using a random single ground truth caption versus multiple ground truth captions at a time.

	MSR-VTT		MSVD	
	R@1	R@5	R@1	R@5
CLIP baseline [55]	32.8	55.7	39.4	64.6
Ours w/ OFA [74]	33.6	59.2	<u>41.1</u>	67.4
Ours w/ ClipCap [46]	34.7	59.8	40.6	68.9
Ours w/ BLIP [32]	**35.8**	60.6	**41.1**	**69.1**
Ours w/ NN-CC	35.4	**61.1**	40.2	66.9

Table A.3: **Retrieving nearest neighbor caption from an image-text dataset:** Instead of using a captioner, we experiment with retrieving the captions from the Conceptual Captions [63] dataset, using the frame embedding as query (NN-CC) and obtain comparable performance to other captioners.

by finetuning our proposed model (Section A.1), and by demonstrating the advantages of MCQS on multi-captioned data (Section A.2).

A.1 Finetuning with ground-truth captions

We show that our proposed methodology can be used as a pretraining step. Here, we experiment with initializing a model trained with automatic captions, and finetuning with ground-truth captions to further improve the performance. Table A.1 summarizes the results. The bottom gray lines compare finetuning the model with ground-truth captions (i) from CLIP initialization (rows with WiT+GT data), or (ii) from pretraining with our method (last row with WiT+GT+PL data). This comparison highlights the benefits of using our proposed methodology as a pretraining step, as it leads to further improvement in performance on the target datasets. We note that when we train with the ground truth, we keep all hyperparameters the same for both (i) finetuning from CLIP initialization or (ii) finetuning from our pretraining with pseudolabels.

A.2 Multi-caption training on MSR-VTT

MSR-VTT videos come with 20 ground-truth captions per video. Therefore, in the fully-supervised setting, we can employ our MCQS approach for training. In Table A.2, we show that using all ground-truth captions at a time with MCQS improves over using a single caption randomly sampled at each training iteration.

B Alternative methods

Retrieving nearest-neighbor caption. One interesting question is whether we need a captioner model to obtain frame captions. Given that there exists a joint space between images and text through CLIP, an alternative approach would be to retrieve the closest text embedding from a large image-text gallery by querying with the video frame embedding (similar in spirit to [48]). We performed this baseline experiment using the Google Conceptual Captions [63] dataset as the image-text gallery source, which was also the ClipCap training set [46]. In a similar fashion to our previous experiments, (i) we extract 10 frames, (ii) retrieve a caption for each frame, and (iii) compute CLIPScore and filter them accordingly. In Table A.3, we show that the retrieved captions can also be used to outperform the zero-shot baseline. However, as will be seen in the next section, the retrieved captions appear to be less similar to the ground truth text than with ClipCap or BLIP (Table A.3).

Captioning bottleneck with text-to-text retrieval. Another baseline we design is to use the captions directly at test time without fine-tuning CLIP. This constitutes an information bottleneck where the video is embedded only into a text, as opposed to a high-dimensional embedding space. To determine the nearest video given a text query, we use the previously extracted captions with ClipCap and BLIP. To represent a given video, (i) we embed the 10 extracted captions with Sentence-BERT [59] (S-BERT), (ii) select

		ActivityNet		MSR-VTT		MSVD	
	Text enc.	R@1	R@5	R@1	R@5	R@1	R@5
CLIP baseline [55]		**23.4**	**49.3**	**32.8**	**55.7**	**39.4**	**64.6**
ClipCap [46]	CLIP	7.7	20.6	13.2	28.5	18.3	34.8
ClipCap [46]	S-BERT	9.3	26.3	16.4	34.3	20.4	44.4
BLIP [46]	CLIP	10.6	28.5	15.8	33.3	25.7	47.6
BLIP [32]	S-BERT	13.1	32.3	18.1	39.0	28.5	52.3

Table A.4: **Captioning bottleneck with text-to-text retrieval:** We experiment with retrieving videos by representing them with the text embedding of the extracted captions. This results in lower performance than the CLIP baseline. We present performances with two different text encoders, the CLIP [55] text encoder and Sentence-BERT [59] (S-BERT). See text for more details.

the two with the highest CLIPScore [25], and (iii) average their embeddings. We then compare a text query (also embedded with S-BERT) with this video representation using cosine similarity. In Table A.4, we summarize the results. Of the two text encodings tested, S-BERT performs better than CLIP text encoder as S-BERT was intentionally trained to detect similar sentences. However, even the best performing caption bottleneck (i.e., BLIP with S-BERT) obtains worse results than the zero-shot CLIP baseline. The poor performance of this caption-based retrieval approach suggests that captions are not sufficient to be used directly for retrieval, but they can instead provide a supervision signal for training.

Text summarization. As mentioned in Section 4.6 of the main paper, we explored using a text summarization model to combine multiple captions in a given video, and our attempts led to inconsistent results, as seen in Table A.5. We experimented with summarizing the 10 captions from the two captioners, (Summ(10C) for ClipCap and Summ(10B) for BLIP) and summarizing the filtered and combined 4 captions (Summ(2C+2B)). To summarize the captions, we use the Ada language model hosted in OpenAI. We empirically find that it helps to prepend a randomly sampled raw caption to the summary, potentially because we obtain a longer caption with both local and global information (i.e., results in Table A.5 improve when the prepend column is not empty, e.g., 37.5 vs 35.9).

C Additional evaluations

In this section, we report cross-dataset evaluations (Section C.1), multi-caption evaluation on ActivityNet (Section C.2), and performance metrics for video-to-text retrieval (Section C.3).

		MSR-VTT		MSVD	
Prepend	Summary	R@1	R@5	R@1	R@5
2C + 2B	-	36.5	61.5	**41.7**	**70.0**
-	Summ(10C)	32.1	58.0	39.4	64.7
10C	Summ(10C)	33.6	58.8	40.3	65.8
-	Summ(10B)	33.7	59.2	40.6	68.0
10B	Summ(10B)	34.4	59.1	41.0	69.0
-	Summ(2C + 2B)	35.9	60.9	40.8	68.8
2C + 2B	Summ(2C + 2B)	**37.5**	**62.2**	38.6	69.4

Table A.5: **Text summarization results:** Results when summarizing the 10 available captions or the Top 2 from each captioner (2C+2B). We explore two variants, training with only the summary (prepend empty), or training with the concatenation of a random caption and the summary. We do not obtain consistent improvements.

Eval	ActivityNet		MSR-VTT		MSVD	
Train	R@1	R@5	R@1	R@5	R@1	R@5
CLIP [55]	23.8	50.0	33.9	57.3	38.5	64.6
ActivityNet	29.7	57.0	38.4	62.7	43.3	69.7
MSR-VTT	29.5	56.7	39.0	64.6	43.5	69.2
MSVD	28.8	55.4	37.9	62.7	42.5	70.0
Combined	**30.6**	**57.9**	**39.2**	**65.1**	**44.5**	**71.8**

Table A.6: **Cross-dataset evaluation:** Diagonal is training and evaluating on the same dataset (Table 5, Self row, in the main paper). Training with MSVD leads to lowest performance (smallest dataset among three). Note that we train three MSVD models with different seeds and report the mean of the recalls.

C.1 Cross-dataset evaluation

As mentioned in Section 4.2 of the main paper, we report cross-dataset evaluations. In Table A.6, we use the models trained with multi-caption query scoring, where the diagonal corresponds to the second-last row of Section 5 (training and evaluating on the same dataset). Interestingly, the performance of MSR-VTT training and evaluating on ActivityNet is almost as good as training with ActivityNet videos. Furthermore, models trained only on MSVD perform poorly on all datasets (including itself), given its small size.

C.2 Multi-caption evaluation on ActivityNet

To evaluate ActivityNet in all the experiments in the paper, we concatenate all the ground-truth captions available for a video and generate a text query as

Method	Eval	ActivityNet R@1	ActivityNet R@5
CLIP [55]	QS	23.8	50.0
Ours (Combined)	QS	30.6	57.9
Ours (Combined)	MCQS	**31.7**	**58.8**

Table A.7: **Multi-caption query-scoring evaluation on ActivityNet:** We compare evaluating with query-scoring (QS) with a single text query per video (concatenating descriptions), with multiple-caption query-scoring.

Method	ActivityNet R@1	ActivityNet R@5	MSR-VTT R@1	MSR-VTT R@5	MSVD R@1	MSVD R@5
CLIP baseline [55]	21.5	45.6	32.3	56.3	35.4	62.4
Ours (Self)	28.5	**56.0**	**36.5**	64.0	40.0	69.7
Ours (Combined)	**28.7**	55.9	36.4	**66.4**	**41.6**	**70.5**

Table A.8: **Video-to-text retrieval metrics:** Our method (2C+2B trained with MCQS and evaluated with QS) also improves the CLIP baseline on video-to-text retrieval metrics.

in [19, 40, 88]. Instead of concatenating the multiple captions to form a single text query, we can use all the available descriptions of a video as text queries and evaluate using our multi-caption query scoring method. In Table A.7, we observe further improvements with this approach.

C.3 Video-to-text retrieval metrics

In the main paper, we only report text-to-video retrieval metrics. Here, in Table A.8, we report the *video*-to-text metrics. We see that our method also improves over the baseline on these metrics.

D Analysis on selecting captions and combining multiple captioners

As mentioned in Section 4.2 of the main paper, we provide further analysis about the source of captions from multiple frames and multiple captioners.

Quantitative results. One way to check the assumption that selecting the best captions is removing noisy captions is to compare the captions with the ground truth. In Table A.9, we compare the extracted captions with the ground truth with two metrics: METEOR and CLIPScore. However, unlike in the main paper, here we compute the CLIPScore between the two texts (extracted and ground-truth captions), rather than between visual and text embeddings. The results motivate the top-2 selection instead of using all 10 captions. We also show the maximum (Max), which corresponds to the comparison of the ground truth with each of the 10 captions individually and selecting the one with the highest score, as a way to give an upperbound on this score assuming a perfect selection method (note that this requires access to the ground truth). We observe that retrieving nearest neighbor captions has the least similarity with the ground-truth text.

	# capt.	ActivityNet M	ActivityNet T-CS	MSR-VTT M	MSR-VTT T-CS	MSVD M	MSVD T-CS
NN	10	-	-	6.9	77.9	7.2	70.1
	Top 2	-	-	7.8	79.6	8.5	72.1
	Max	-	-	12.0	85.9	14.9	78.4
C	10	15.3	70.3	9.1	81.9	9.3	73.0
	Top 2	16.6	71.2	10.4	82.2	10.7	74.2
	Max	26.1	78.4	14.9	88.6	17.9	80.6
B	10	17.2	74.0	20.2	85.6	21.0	79.0
	Top 2	17.7	74.4	20.5	86.4	21.7	79.8
	Max	28.0	80.4	27.5	91.9	31.8	84.6

Table A.9: **Comparing automatic captions to ground-truth text:** We compare the extracted captions from Nearest Neighbour (NN), ClipCap (C) and BLIP (B) approaches to ground-truth video captions, with METEOR (M) [6] and Text CLIPScore (T-CS) [25] metrics. When we evaluate 10 or 2 captions, we compute the metrics individually for each caption and report the average. For the maximum, we compute the metrics for all the 10 captions and select the one with the highest score. Retrieving nearest neighbour captions have the least similarity with the ground truth text. Filtering captions with CLIPScore (Top 2) improves all metrics.

Different CLIPScore distributions. As seen in Figure A.1, ClipCap and BLIP captions have different CLIPScore distributions, with μ being higher for ClipCap, perhaps due to the CLIP backbone. If we were to select the best 4 captions out of the 20 available ones, we would be selecting ClipCap captions more often than BLIP captions.

Top 4 of all the captions. We see in Figure A.2 that combining captions from different captioners is better than using only ClipCap or BLIP. Out of the two alternatives: (i) selecting top 4 of the 20 combined set of captions, (ii) selecting top 2 from each captioner, option (ii) leads to better results.

Number of different frames. When we select Top 2 from one captioner, our captions come from only two frames. In Table A.10, we see statistics of the amount of different frames when combining Top 2 of ClipCap

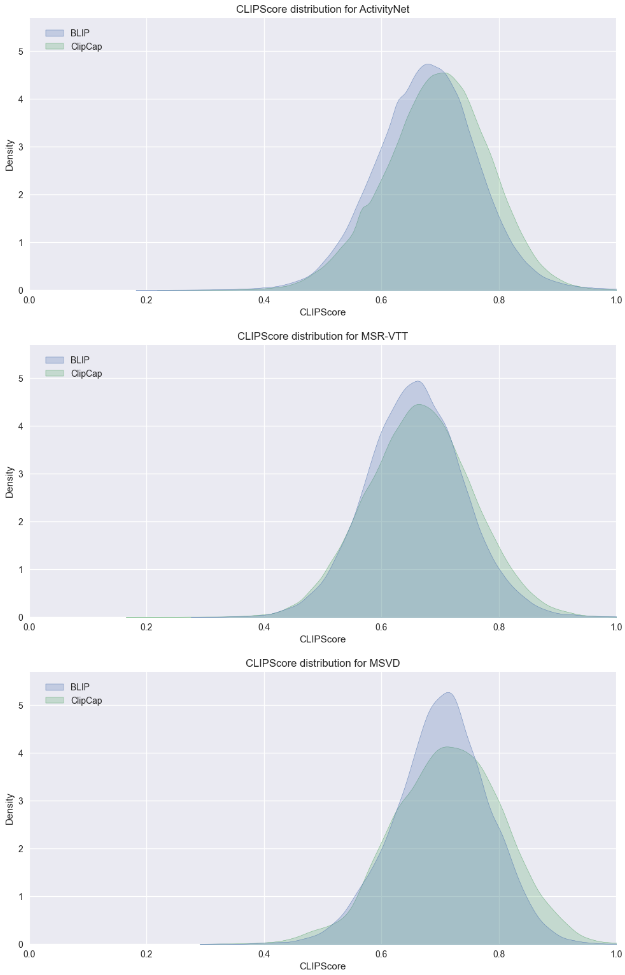


Fig. A.1: **CLIPScore kernel density estimate:** We plot the CLIPScore distribution for three datasets, and both models (ClipCap and BLIP). CLIPScore is higher for ClipCap than for BLIP, potentially because of the CLIP backbone.

with Top 2 of BLIP. It can be seen that only around 7% of the time the top captions from both captioners come from the exact two frames. More than 44% of the time there is a frame in common with the two captioners. Finally, most frequently, 4 *different* frames are selected from the 10 possible frames: 2 from each captioner.

Repetitive captions. One other benefit of filtering the captions is that we are left with a set of less repetitive captions. See Figure A.3 for the percentage of unique captions when using 10 captions and Top 2 captions. We also check that there are less than 1% of overlapping captions between the two captioners in any of the three datasets. This is yet another reason that motivates us to use different captioners and obtain more diverse and rich captions.

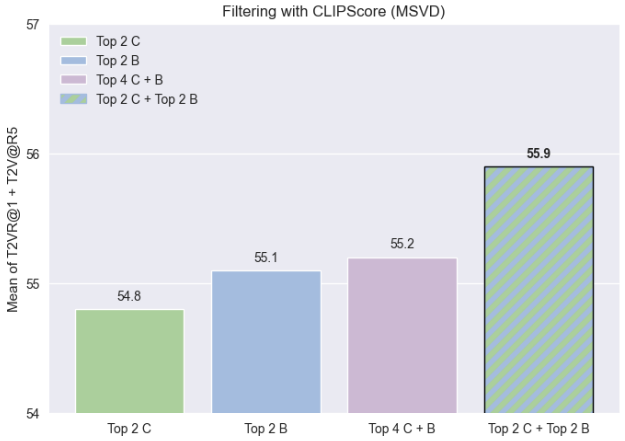


Fig. A.2: **Combining captioners:** We compare 4 different strategies: selecting 2 from 10 ClipCap captions, selecting 2 from 10 BLIP captions, selecting Top 4 from the 20 combined captions, selecting Top 2 from each captioner. We highlight the best performance with a black border.

Dataset	4 frames	3 frames	2 frames
ActivityNet	47.4%	45.4%	7.2%
MSR-VTT	48.5%	44.2%	7.3%
MSVD	47.4%	44.8%	7.8%

Table A.10: **Different frames:** When using C+B Top-2 (4 captions), about 47% of the videos have captions from 4 different frames, and around 45% of the videos have captions from 3 different frames (i.e., the two captioners pick the same one frame in their top rankings). Finally, there are roughly 7% of the videos where both captioners select the same two frames. In these cases, multiple captions can still be useful to provide data augmentation.

	Caption	Temporal pooling		MSVD	
Captioners	pooling	train	eval	R@1	R@5
CLIP baseline [55]		-	mean	39.4	64.5
		-	QS	38.5	64.5
C + B	Rand(4)	mean	mean	41.7	**70.0**
C + B	Mean(4)	MCQS	QS	42.5	**70.0**
C + B + O	Rand(6)	mean	mean	41.8	69.2
C + B + O	Mean(6)	MCQS	QS	**42.8**	68.5

Table A.11: **Combining with OFA:** We experiment with combining three captioners, i.e., using a total of 6 captions by selecting top 2 from each of the ClipCap (C), BLIP (B), and OFA (O) captioners. While the R@1 metric improves when adding a third captioner, we see no further improvement in R@5.

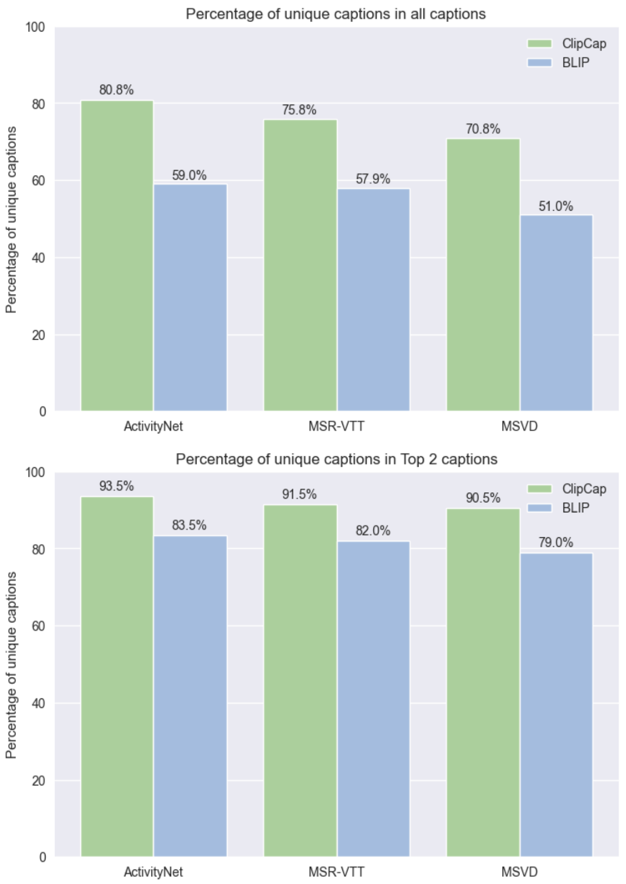


Fig. A.3: **Percentage of unique captions:** We make statistics about the percentage of unique extracted captions within a video (top: for all 10 captions, bottom: for the best 2 captions). We observe that BLIP captions are more diverse, and ClipCap ones are a bit more repetitive.

Beyond two captioners. We explore using three different captions by combining ClipCap (C), BLIP (B) and OFA (O) in Table A.11. The results do not bring consistent improvements in both metrics (better R@1, worse R@5), possibly because OFA performance alone is not as effective compared to BLIP.

E Implementation details for the BLIP initialization experiment

We here explain the BLIP implementation details of the backbone experiments in Section 6. We train using a method akin to that of BLIP, where the Image-Text Contrastive (ITC) loss is denoted as our L in Eq. (5). For the Image-Text Matching (ITM) loss, we extend the encoder hidden states by the number of frames. We train with 4 frames and evaluate with 8 frames. We adopt the ViT-B/16 backbone for the image encoder and the BERT architecture [14] for the text encoder as in BLIP. We train the model with a single NVIDIA RTX A600 using 4 frames, while evaluations are conducted using 8 frames as in the original paper.

F Additional qualitative results

Captioning. Similar to Figure 2 of the main paper, in Figure A.4, we provide more examples of captioning results from both ClipCap and BLIP, together with their corresponding CLIPScores when compared to the image embeddings. In the third picture of the second video or in the first picture of the third video, we see that CLIPScore is low when the captions does not match the frame. In the last video, we see examples of a short video where all the frames look alike, and the extracted captions are the same or almost the same.
Retrieval. To complement Figure 3 of the main paper, we provide additional qualitative results in Figure A.5 for the three datasets: ActivityNet (first two rows), MSR-VTT (middle two rows) and MSVD (last two rows).

G Data availability statement

We conducted experiments using three popular text-to-video retrieval public datasets, namely ActivityNet [29], MSR-VTT [78], and MSVD [9]. The URLs to download the datasets are:

- ActivityNet
- MSR-VTT
- MSVD

We complement them with our automatic caption labels and will release these along with our code and pretrained models.

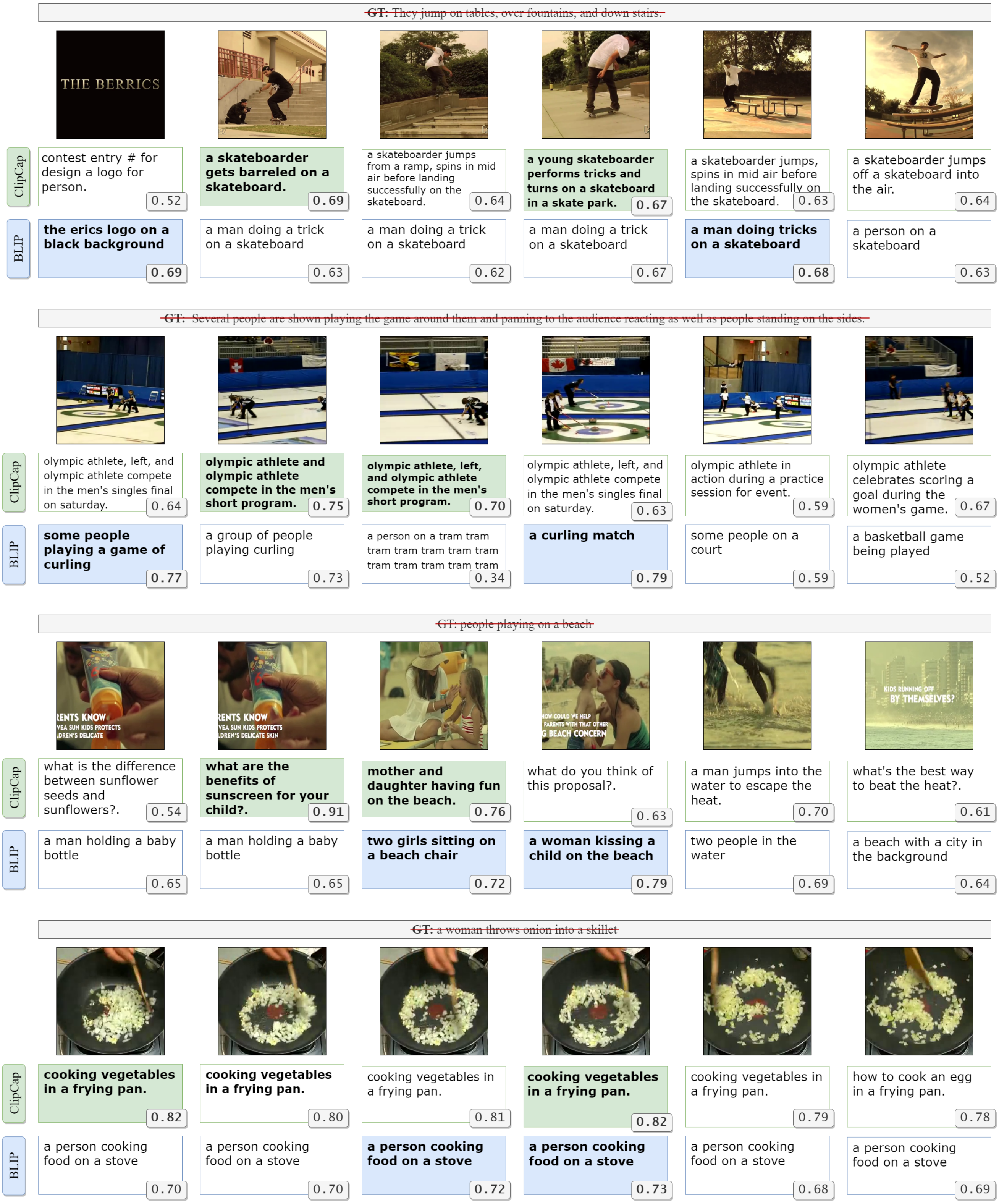
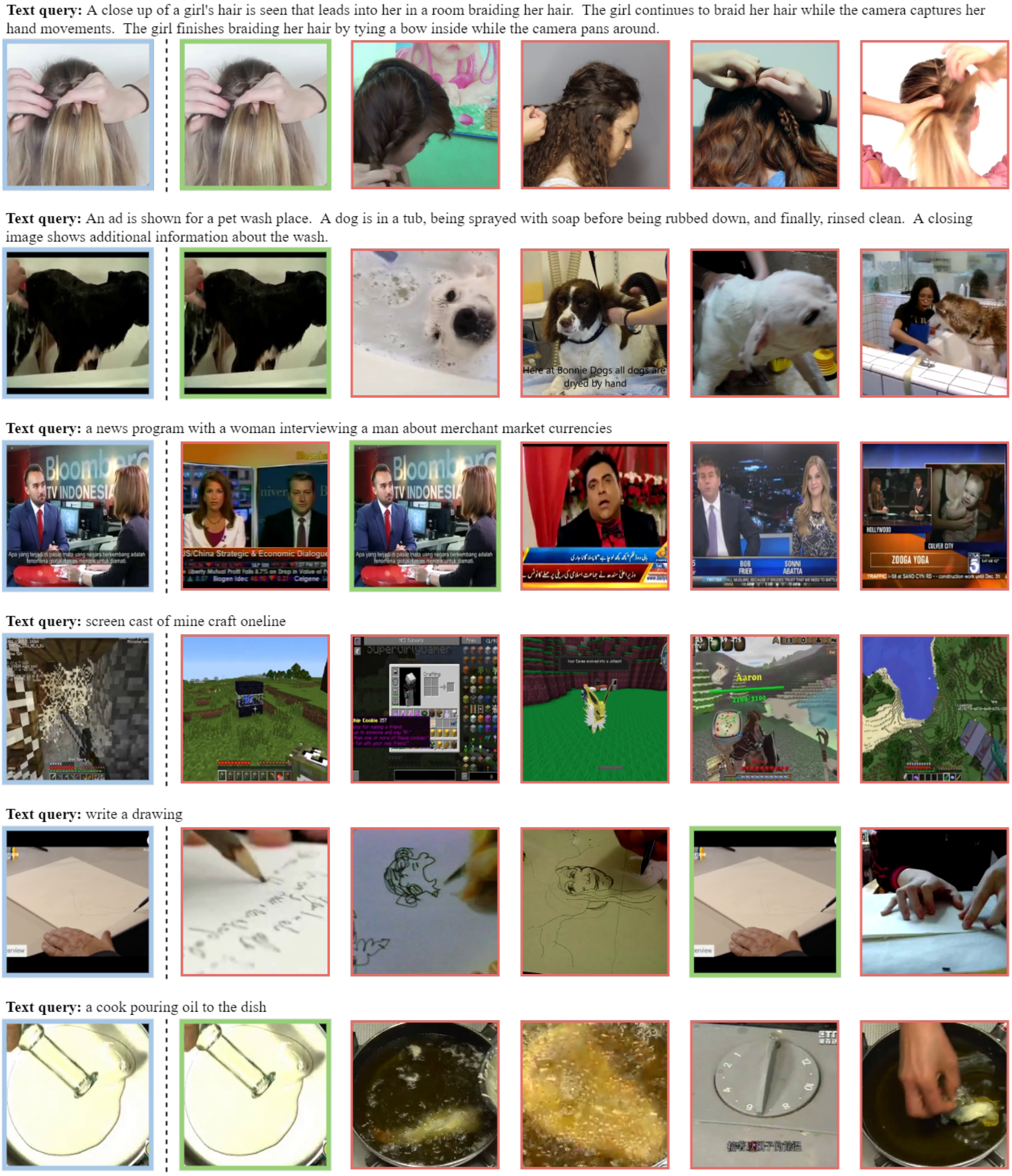

~~GT: They jump on tables, over fountains, and down stairs.~~
THE BERRICS
ClipCap
contest entry # for design a logo for person. 0.52
a skateboarder gets barreled on a skateboard. 0.69
a skateboarder jumps from a ramp, spins in mid air before landing successfully on the skateboard. 0.64
a young skateboarder performs tricks and turns on a skateboard in a skate park. 0.67
a skateboarder jumps, spins in mid air before landing successfully on the skateboard. 0.63
a skateboarder jumps off a skateboard into the air. 0.64
BLIP
the erics logo on a black background 0.69
a man doing a trick on a skateboard 0.63
a man doing a trick on a skateboard 0.62
a man doing a trick on a skateboard 0.67
a man doing tricks on a skateboard 0.68
a person on a skateboard 0.63
~~GT: Several people are shown playing the game around them and panning to the audience reacting as well as people standing on the sides.~~
ClipCap
olympic athlete, left, and olympic athlete compete in the men's singles final on saturday. 0.64
olympic athlete and olympic athlete compete in the men's short program. 0.75
olympic athlete, left, and olympic athlete compete in the men's short program. 0.70
olympic athlete, left, and olympic athlete compete in the men's singles final on saturday. 0.63
olympic athlete in action during a practice session for event. 0.59
olympic athlete celebrates scoring a goal during the women's game. 0.67
BLIP
some people playing a game of curling 0.77
a group of people playing curling 0.73
a person on a tram tram tram tram tram tram tram tram tram tram tram tram 0.34
a curling match 0.79
some people on a court 0.59
a basketball game being played 0.52
~~GT: people playing on a beach~~
ClipCap
what is the difference between sunflower seeds and sunflowers?. 0.54
what are the benefits of sunscreen for your child?. 0.91
mother and daughter having fun on the beach. 0.76
what do you think of this proposal?. 0.63
a man jumps into the water to escape the heat. 0.70
what's the best way to beat the heat?. 0.61
BLIP
a man holding a baby bottle 0.65
a man holding a baby bottle 0.65
two girls sitting on a beach chair 0.72
a woman kissing a child on the beach 0.79
two people in the water 0.69
a beach with a city in the background 0.64
~~GT: a woman throws onion into a skillet~~
ClipCap
cooking vegetables in a frying pan. 0.82
cooking vegetables in a frying pan. 0.80
cooking vegetables in a frying pan. 0.81
cooking vegetables in a frying pan. 0.82
cooking vegetables in a frying pan. 0.79
how to cook an egg in a frying pan. 0.78
BLIP
a person cooking food on a stove 0.70
a person cooking food on a stove 0.70
a person cooking food on a stove 0.72
a person cooking food on a stove 0.73
a person cooking food on a stove 0.68
a person cooking food on a stove 0.69

Fig. A.4: **Qualitative results for captioning:** We visualize further captioning results by both ClipCap and BLIP as in Figure 2 of the main paper. First two rows are on ActivityNet, third row on MSR-VTT, last row on MSVD.



Text query: A close up of a girl's hair is seen that leads into her in a room braiding her hair. The girl continues to braid her hair while the camera captures her hand movements. The girl finishes braiding her hair by tying a bow inside while the camera pans around.

Text query: An ad is shown for a pet wash place. A dog is in a tub, being sprayed with soap before being rubbed down, and finally, rinsed clean. A closing image shows additional information about the wash.

Text query: a news program with a woman interviewing a man about merchant market currencies

Text query: screen cast of mine craft oneline

Text query: write a drawing

Text query: a cook pouring oil to the dish

Fig. A.5: **Qualitative text-to-video retrieval results:** Above, video retrieval results for our best model (Combined) are shown. The examples belong to the test sets of ActivityNet (first two rows), MSR-VTT (third and fourth rows), and MSVD (last two rows). Each example is shown with the text query, the ground-truth video (first column, blue border), and the top 5 retrieved videos from the gallery. Every video is only displayed using the middle frame, with a green border if it matches the ground-truth video, or a red border otherwise. Overall, all the retrieved videos have similar semantic meaning with the text query, even in cases where the correct video is not retrieved at the first rank.