TEACHTEXT: CrossModal Generalized Distillation for Text-Video Retrieval

Ioana Croitoru [1,2,*] Simion-Vlad Bogolin[1,2,*] Marius Leordeanu[2,3]
Hailin Jin[4] Andrew Zisserman[1] Samuel Albanie[1,5,†] Yang Liu[1,6,†]
[1]Visual Geometry Group, Univ. of Oxford [2]Inst. of Mathematics of the Romanian Academy [3]Univ. Politehnica of Bucharest
[4]Adobe Research [5]Dept. of Engineering, Univ. of Cambridge [6]Wangxuan Inst. of Computer Technology, Peking Univ.

Abstract

In recent years, considerable progress on the task of text-video retrieval has been achieved by leveraging large-scale pretraining on visual and audio datasets to construct powerful video encoders. By contrast, despite the natural symmetry, the design of effective algorithms for exploiting large-scale language pretraining remains under-explored. In this work, we are the first to investigate the design of such algorithms and propose a novel generalized distillation method, TEACHTEXT*, which leverages complementary cues from multiple text encoders to provide an enhanced supervisory signal to the retrieval model. Moreover, we extend our method to video side modalities and show that we can effectively reduce the number of used modalities at test time without compromising performance. Our approach advances the state of the art on several video retrieval benchmarks by a significant margin and adds no computational overhead at test time. Last but not least, we show an effective application of our method for eliminating noise from retrieval datasets. Code and data can be found at* `https://www.robots.ox.ac.uk/~vgg/research/teachtext/`.

1. Introduction

The focus of this work is *text-video retrieval*—the task of identifying which video among a pool of candidates best matches a natural language query describing its content. Video search has a broad range of applications across domains such as wildlife monitoring, security, industrial process monitoring and entertainment. Moreover, as humanity continues to produce video at ever-increasing scale, the ability to perform such searches effectively and efficiently takes on critical commercial significance to video hosting platforms such as YouTube.

A central theme of recently proposed retrieval methods has been the investigation of how to best use multiple video modalities to improve performance. In particular, architectures based on mixtures-of-experts [28, 32] and multi-modal transformers [18] have shown the benefit of making use of diverse sets of pre-trained models for related tasks (such as image classification, action recognition and ambient sound classification) as a basis for video encoding during training and testing.

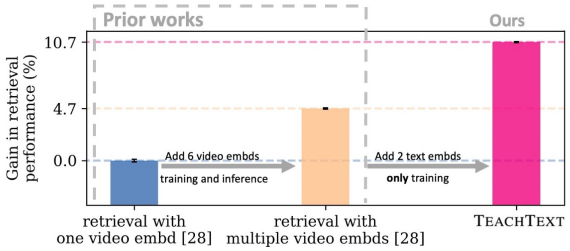

Figure 1. **Distilling the knowledge from multiple text encoders for stronger text-video retrieval.** Prior works [18, 28, 32] have shown the considerable benefit of transitioning from video encoders that ingest a single modality (*left*) to multi-modal video encoders (*centre*). In this work, we show that retrieval performance can be further significantly enhanced by learning from multiple text encoders through the TEACHTEXT algorithm which imposes no additional cost during inference. Text-to-video retrieval performance gain (geometric mean of R1-R5-R10) is reported for a [28] model as well as for our method on the MSR-VTT [55] dataset.

In this work, we explore whether commensurate gains could be achieved by leveraging multiple text embeddings learned on large-scale written corpora. Different from video embeddings using multiple modalities and pretraining tasks, it is less obvious that there is sufficient diversity among collections of text embeddings to achieve a meaningful boost in performance. In fact, our inspiration stems from a careful investigation of the performance of different text embeddings across a range of retrieval benchmarks (Fig. 2). Strikingly, we observe not only that there is considerable variance in performance across text embeddings, but also that *their ranking is not consistent*, strongly supporting the idea of using multiple text embeddings.

Motivated by this finding, we propose a simple algorithm, TEACHTEXT, to effectively exploit the knowledge

[*]Equal contribution. [†]Corresponding authors.

captured by collections of text embeddings. Our approach requires a "student" model to learn from a single or multiple "teacher" retrieval models with access to different text embeddings by distilling their text-video similarity matrices into an enhanced supervisory signal. As shown in Fig. 1, TEACHTEXT is capable of delivering a significant performance gain. Moreover, this gain is complementary to that of adding more video modalities to the video encoder but importantly, unlike the addition of video modalities, does not incur additional computational cost during inference.

Our main contributions can be summarised as follows: (1) We propose the TEACHTEXT algorithm, which leverages the additional information given by the use of multiple text encoders; (2) We show that directly learning the retrieval similarity matrix between the joint query video embeddings, which to the best of our knowledge is novel, is an effective generalized distillation technique for this task (and we compare our approach to alternatives among prior work such as uni-modal relationship distillation [37]); (3) We show an application of our approach in eliminating noise from modern training datasets for the text-video retrieval task; (4) We demonstrate the effectiveness of our approach empirically, achieving state of the art performance on six text-video retrieval benchmarks.

2. Related Work

Video retrieval methods. The task of indexing video content to enable retrieval has a rich history in computer vision—sophisticated systems have been developed to find specific objects [45], actions [26], predefined semantic categories [21], irregularities [4] and near-duplicates [13, 44]. In this work, we focus on the task of retrieving content that matches a given natural language description. For this particular task, there has been considerable interest in developing cross-modal methods that employ a joint-embedding space for text queries and video content [2, 3, 15, 35, 54, 56, 57]. These joint video-text embeddings, which aim to map videos and text descriptions into a common space such that matching video and text pairs are close together, form an attractive computational model for tackling this problem, since they allow for efficient indexing (although hierarchical embeddings have also been investigated [12]). Recently, two key themes have emerged towards improving the quality of these embeddings. First, large-scale weakly supervised pretraining methods [24, 31, 33] have sought to expand their training data by exploiting the speech contained in the videos themselves as a supervisory signal. Second, the integration of multiple modalities (which has long been considered important for semantic indexing [46]) has been shown to yield significant gains in performance [18, 28, 32, 39]. We focus on candidates from this latter theme as a basis for investigating our approach.

Text embeddings. The representation of language through learned embeddings has been widely studied [34, 40, 41] and applied in a variety of natural language processing applications. Several works have demonstrated that even with large-scale pretraining, there still are benefits to finetuning the models on the target task [14, 40] and that larger models (often employing multiple attention heads) yield higher performance [14]. Recently, [8] provided a detailed comparisons on the importance of language features for vision applications and proposes a word embedding that is specifically designed for vision tasks. In this work, we first study how various pretrained language embeddings affect the performance for text-video retrieval and then propose a method to take advantage of the benefits of combining multiple text embeddings.

Knowledge Distillation/Privileged Information. The purpose of knowledge distillation is to transfer knowledge from one model (teacher) to another model (student). This idea was originally introduced in the context of decision tree simplification [6] and model compression [7], and later extended by [19] who formalised this knowledge transfer as the temperature-parameterised process of *knowledge distillation*. The concept was further generalised in the unifying framework of *generalized distillation* [30] for learning with privileged information [50] (via *similarity control* and *knowledge transfer* [49]), together with knowledge distillation [19]. Our approach distills knowledge of the similarities between video and text samples into the student and therefore represents a form of generalized distillation. While most knowledge distillation methods train the student with the teacher's outputs as targets, more recent methods propose different approaches [20, 43, 58]. Of most relevance to our approach, [37] transfer mutual relations of data examples and propose distance-wise and angle-wise distillation losses that penalize structural differences in relations instead of training the student to mimic the output of the teacher—we compare to their approach in Sec. 5.

3. Motivation and intuition

Recently, [41] points out that even though language representation learning systems (such as [25, 29, 40]) are pretrained on vast amounts of data, they are still sensitive to slight changes in the data distribution and task specification. In this way, most systems can be viewed as *narrow experts rather than competent generalists*.

Consequently, in Fig. 2 we investigate how the usage of different off-the-shelf pre-trained text embeddings affects the retrieval performance. We observe that there is significant variance both within and across datasets, suggesting that each embedding captures different types of information. Our intuition is that this information comes from the diversity of architectures, pretraining datasets and pretraining objectives, which differs across the text embeddings.

Next, we give details about the used text embeddings

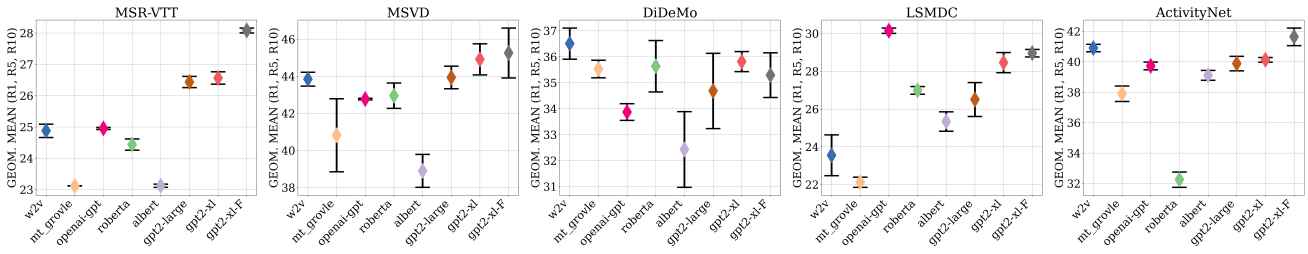

Figure 2. **Influence of varying the text embedding.** Different text embeddings are presented on the x axis: w2v [34], mt_grovle [8], openai-gpt [40], roberta [29], albert [25], gpt2-large [41], gpt2-xl [41], gpt2-xl-F along with their performance in geometric mean of $R1$-$R5$-$R10$ on five datasets. For each experiment, we report the mean (diamond) and standard deviation (error bar) of three randomly seeded runs.This study is performed using the CE retrieval architecture [28]: each model differs only in its use of pre-trained text embedding at input. We observe a significant variance in performance when changing the text embedding, both across and within datasets. The difference in rankings across datasets suggests the presence of additional information among different text embeddings.

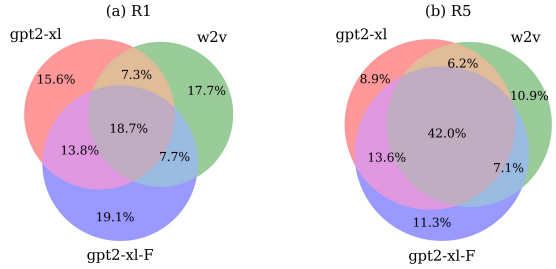

Figure 3. **Share of correctly retrieved samples based on the used pre-trained text embedding on MSR-VTT.** We observe that each embedding has a considerable share of sample retrieved correctly only by itself (in terms of R1 left and R5 right), further justifying our approach. Best viewed in color.

and summarise the key differences between them in relationship with our findings. Word2vec (w2v) [34] is a lightweight text embedding that is widely used for vision tasks [10, 27, 52]. Multi-task GrOVLE (mt_grovle) [8], is an extension of w2v that is specially designed for vision-language tasks (in our experiments, however, we find that it slightly under-performs w2v). The finetuned transformer language model (openai-gpt) [40] embedding is trained on a book corpus containing long stretches of contiguous text. We observe that it performs well on datasets that have longer text queries such as ActivityNet. RoBERTa and ALBERT [25, 29] are based on the BERT architecture [14] and are trained on the same data which consists of unpublished books and Wikipedia articles. RoBERTa [29] focuses on hyperparameter optimization and shows that greater model capacity leads to better performance while ALBERT[25] proposes some parameter-reduction techniques to reduce memory consumption and increase training speed. In our experiments, we observe a high variation in performance when comparing the two. In contrast to the other embeddings, gpt2[41] is trained on a crawled dataset that was designed to be as diverse as possible. We observe that gpt2 performs most robustly in our experiments, especially on smaller datasets such as MSR-VTT and MSVD. However, it nevertheless exhibits a domain gap to each corpus (highlighted by the fact that performance increases when fine-tuning gpt2-xl, termed gpt2-xl-F throughout the paper, on queries from the text-video retrieval datasets).

Additionally, in Fig. 3 we show how many correctly retrieved queries are shared between three text embeddings on MSR-VTT: gpt2-xl, gpt2-xl-F and w2v. Only around 19% (R1), respectively 42% (R5) queries are correctly retrieved by all the three considered text embeddings. This means that a significantly number of queries are sensitive to the used text embedding, consolidating our intuition.

4. Method

Motivated by the findings from Sec. 3, our work aims to study the influence of using multiple text embeddings for text-video retrieval.

4.1. Problem description and learning setup

Let $D = \{(v_i, c_i)\}_{i=1}^{n}$ be a dataset of paired videos and captions. Following the multi-modal experts approach of [18, 28, 32], for each video we have access to a collection of video embeddings (sometimes referred to as "experts") x_i extracted from the various modalities of video v_i using a pretrained video encoder (*VE*) in addition to a text embedding t_i (extracted using a text encoder, *TE*) for each caption/query c_i[1]. The objective of the text-video retrieval task is to learn a model $M(x_i, t_j)$ which assigns a high similarity value to pairings (x_i, t_j) of video and text embeddings that are in correspondence (i.e. $i = j$) and a low similarity otherwise. As is common in the literature [5, 32], we parameterise the model as a dual-encoder that produces joint-embeddings in a shared space such that they can be compared directly $M(x_i, t_j) = F(x_i)^T Q(t_j) \in \mathbb{R}$ where F and Q represent the learnt video and text encoder respectively. To train the video and text encoder for the task of

[1]These embeddings are produced by models that have been trained on relevant tasks (such as action recognition for the video encoder and language modelling for the text encoder)

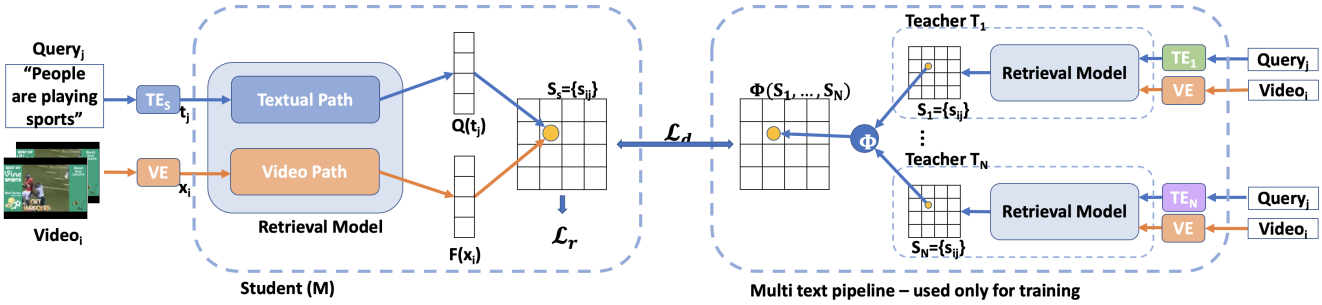

Figure 4. **TEACHTEXT teacher-student framework overview.** Given a batch of input videos and queries in natural language during training, the student model, M (left) and teacher models $T_1, \ldots, T_N$ (right) each produce similarity matrices (visualised as square grids). The similarity matrix produced by M is encouraged to match the aggregated matrices of the teachers through the distillation loss $\mathcal{L}_d$ in addition to the retrieval loss $\mathcal{L}_r$. Note that both the student and teachers ingest the same video embeddings (VE), but employ different text embeddings (TE_S for the student, $\text{TE}_1, \ldots, \text{TE}_N$ for the teachers). At test time, the teacher models are discarded.

retrieval, we adopt a contrastive ranking loss [47]:

$$\mathcal{L}_r = \frac{1}{B}\sum_{i=1}^{B}\sum_{i\neq j}[max(0, s_{ij} - s_{ii} + m) + max(0, s_{ji} - s_{ii} + m)] \quad (1)$$

where B represents the batch size used during training, $s_{ij} = F(x_i)^T Q(t_j)$ is the similarity score between the encoded video $F(x_i)$ and query $Q(t_j)$ while m is the margin.

The key idea behind our approach is to learn a retrieval model, M, that, in addition to the loss described above, also has access to information provided by a collection of pre-trained "teacher" retrieval models which are trained on the same task but ingest different text embeddings.

4.2. TEACHTEXT algorithm

To enhance the retrieval performance of model M, we propose the TEACHTEXT algorithm which aims to exploit cues from multiple text embeddings. An overview of our approach is provided in Fig. 4. Initially, we train a collection of teacher models $\{T_k : k \in \{1, \ldots, N\}\}$ for the text-video retrieval task using the approach described in Sec. 4.1. The teachers share the same architecture but each model T_k uses a different text embedding as input (extracted using a pre-trained text encoder TE_k). In the second phase the parameters of the teachers are frozen. We then proceed by sampling a batch of B pairs of videos and captions and computing a corresponding similarity matrix $S_k \in \mathbb{R}^{B\times B}$ for each teacher T_k (Fig. 4 right). These N similarity matrices are then combined with an aggregation function, $\Phi : \mathbb{R}^{N\times B\times B} \rightarrow \mathbb{R}^{B\times B}$, to form a single supervisory similarity matrix (Fig. 4, centre-right). Concurrently, the batch of videos and captions are likewise processed by the student model, M, which produces another similarity matrix, $S_s \in \mathbb{R}^{B\times B}$. Finally, in addition to the standard retrieval loss (Eq. 1), a distillation loss, $\mathcal{L}_d$, encourages the S_s to lie close to the aggregate $\Phi(S_1, \ldots, S_N)$. The algorithm is summarized in Alg. 1. During inference, the teacher models are discarded and the student model M requires only a single text embedding. Next, we give details of the distillation loss used for the similarity matrix learning.

Algorithm 1 TEACHTEXT algorithm

Phase 1: *Learn teacher models*
  Train N teacher models $T_k = (F_k, Q_k)$, $k \in \{1, \ldots, N\}$ using the training pairs (x_i, t_j^k) where t_j^k represents the text modality used by teacher T_k in a standard retrieval training setup (Sec. 4.1).
Phase 2: *Learn the student model*, $M = (F, Q)$
  for minibatch of B paired samples $\{(v_i, c_i)\}$ **do**
    For each pair (v_i, c_i) extract video experts and text embedding pairs (x_i, t_i) using *VE* and TE_S.
    Compute student similarity matrix S_s where $S_s(i,j) = F(x_i)^T Q(t_j)$ for $i, j \in \{1, \ldots, B\}$
    Compute the loss $\mathcal{L}_r$ via Eqn. 1 using S_s.
    for teacher T_k, $k = 1, \ldots, N$ **do**
      For each pair (v_i, c_i) extract the video experts and text embedding pairs (x_i, t_i^k) using VE and TE_k.
      Compute the similarity matrix S_k where $S_k(i,j) = F_k(x_i)^T Q_k(t_i^k)$ for $i, j \in \{1, \ldots, B\}$.
    end for
    Compute aggregate teacher matrix $\Phi(S_1, \ldots, S_N)$.
    Compute the loss $\mathcal{L}_d$ between S_s and $\Phi(S_1, \ldots, S_N)$ via Eqn. 2.
    Update M with gradients computed from the composite loss $\mathcal{L} = \mathcal{L}_r + \mathcal{L}_d$.
  end for

4.3. Learning the similarity matrix

As noted in Sec. 4.1, the essence of the retrieval task is to create a model that is able to establish cross-modal correspondences between videos and texts/queries, assigning

a high similarity value to a pairing in which a query accurately describes a video, and a low similarity otherwise. This renders the similarity matrix a rich source of information about the knowledge held by the model. In order to be able to transfer knowledge from the teachers to the student, we encourage the student to produce a similarity matrix that matches an aggregate of those produced by the teachers. In this way, we convey information about texts and video correspondences without strictly forcing the student to produce exactly the same embeddings as the teachers. To this end, we define the similarity matrix distillation loss as:

$$\mathcal{L}_d = \frac{1}{B}\sum_{i=1}^{B}\sum_{j=1}^{B}[l(\Phi(i,j), S_s(i,j))] \tag{2}$$

where B represents the batch size, $\Phi = \Phi(S_1, \ldots, S_N)$ represents the aggregate of the teacher similarity matrices and S_s represents the similarity matrix of the student. Finally, inspired from other distillation works such as [37], l represents the Huber loss and is defined as

$$l(x,y) = \begin{cases} \frac{1}{2}(x-y)^2 & \text{if } |x-y| \leq 1, \\ |x-y| - \frac{1}{2} & otherwise \end{cases} \tag{3}$$

We explored several forms of aggregation function and found that a simple element-wise mean, $\Phi(S_1, \ldots, S_N) = \frac{1}{N}\sum_{k=1}^{N} S_k$, worked well in practice.

The idea of learning directly the cross-modal similarity matrix is, to the best of our knowledge novel. It draws inspiration from the work of relational knowledge distillation [37] which considered the idea of learning from relationships and introduced two algorithms to implement this concept in a uni-modal setting through pairwise and triplet distance sampling. We compare our matrix learning approach with theirs in Sec. 5.

4.4. Student model

A key advantage of our approach is that it is agnostic to the architectural form of the student and teachers, and thus the student (and teachers) can employ any method from the current literature. We test our TEACHTEXT algorithm using three different recent works MoEE [32], CE [28], MMT [18] as the student and teacher base architectures. All these works employ multi-modal video encoders for the text-video retrieval task. For more details, please consult the original paper of each method.

Establishing a stronger baseline. In addition to these models, we also investigate our approach on a model which shares the CE architecture of [28] but includes a series of small technical improvements to provide a stronger baseline against which we also test the TEACHTEXT algorithm. Starting from this base architecture, we refine the input embedding selection, finding that the face and OCR video modalities employed by [28] do not consistently produce improvement so we remove them as inputs to the video encoder. We update the model to use the more powerful gpt2-xl text embedding of [41] and following [18], we finetune this text embedding on captions from the target dataset to bring additional improvement. Combining all of these changes (ablations provided in Sec. 5.3 and Fig. 5a) results in the CE+ model which we include as an additional baseline. Thus, in summary we use four ([18, 28, 32] and CE+) different base architectures for the student model.

4.5. Teacher models

The teacher models use the same architecture as the student model. Concretely, for each of the four base architectures described in Sec. 4.4, we create a pool of multiple teachers, each using a different pre-trained text embedding as input. The candidate text embeddings we consider are: mt_grovle [8], openai-gpt [40], gpt2-large [41], gpt2-xl [41], w2v [34]. So, we obtain a set of up to five models that form the teachers T_k, $k = 1..5$ used by TEACHTEXT.

4.6. Training and implementation details

In order to train our final student, we combine the retrieval loss and the proposed distillation loss $\mathcal{L} = \mathcal{L}_r + \mathcal{L}_d$. Our model is trained in Pytorch [38] using the Adam [22] optimizer. TEACHTEXT does not add any additional trainable parameters or modalities to the final model. Moreover, when training the student using TEACHTEXT, only the additional loss term $\mathcal{L}_d$ is added, all other hyper-parameters remaining the same.

5. Experimental setup

5.1. Datasets description

To provide an extensive comparison we test our approach on seven video datasets that have been explored in recent works as benchmarks for the task of text-video retrieval: **LSMDC** [42], **DiDeMo** [1], **MSVD** [11], **MSRVTT** [55], **ActivityNet** [9], **VaTeX** [53] and **QuerYD** [36]. We follow the same experimental setup as prior works [12, 18, 28, 39].

5.2. Metrics

To assess performance, we follow prior work (e.g [15, 18, 28, 32, 33, 35, 57]) and report standard retrieval metrics, including R@K (recall at rank K, where higher is better) and MdR (median rank where lower is better). For certain analyses, to maintain conciseness we report the geometric mean of R@1, R@5 and R@10 rather than individual metrics (this statistic aims to be representative of overall retrieval performance). The numbers are reported for the task of retrieving a video given text queries `t2v` which is more common in real world applications. The numbers for the reverse task `v2t` and the number of parameters for each model are reported in the Suppl. Mat. For each experi-

ment, we report the mean and standard deviation of three randomly seeded runs.

5.3. Ablations

In this section we present an extensive study of our proposed approach. Following the setup used in prior works [18, 28] we conduct ablations on the MSR-VTT dataset [55], except where otherwise stated.

Baseline improvements. We propose CE+ as an additional baseline which consists of a series of technical improvements to the model of [28]. As seen in Fig. 5a each modification described in Sec. 4.4 brings additional gain over the base architecture. We observe in particular that finetuning the text embedding on the target dataset has a high influence, further highlighting the critical role played by text embeddings and justifying their study. In addition to other changes we found that certain video embedding expert features were highly sensitive to compression choices used in video pre-processing, which we correct accordingly (more details in Suppl. Mat.). Please note that for a fair comparison, in Sec. 5.4 we report the numbers of re-training the methods [28, 32] using these embeddings extracted with the updated pre-processing which yields a higher performance than the ones reported in the original papers.

Using multiple text embeddings during inference. TEACHTEXT makes no use of additional information at test time. However, it is natural to ask whether the additional text embeddings can be trivially included as part of the model architecture. In Fig. 5(b) we compare our approach with some relatively simple text embedding aggregation techniques, which require access to multiple text embeddings during both training and inference. We observe that TEACHTEXT outperforms these aggregation techniques such as direct concatenation or mean of the text embeddings, suggesting that the proposed method is effective in capturing the additional information given by multiple text embeddings. Moreover, the text encoder of existing systems [18, 28, 32] typically employs many parameters, so adding multiple text embeddings to the architecture adds a significant number of parameters (100M+). For example, the concatenation of two text embeddings (provided that they have the same size) almost doubles the total number of parameters for CE+. In contrast, when employing TEACHTEXT, no parameters are added.

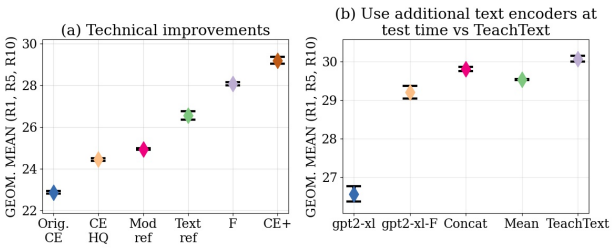

Figure 5. **(a) Baseline improvements**. The y-axes (scaled for clarity) denotes retrieval performance on MSR-VTT. We begin by presenting the performance of the original CE [28]. Firstly, we correct compression artefacts in the pre-processing used for embedding extraction (*CE HQ*, more details in Suppl. Mat.). Secondly, we refine the used video modalities and text modalities (*Mod ref* and *Text ref*). Finally, we finetune the text embedding (*F*) and change the optimizer to Adam [22], thus obtaining the *CE+* baseline. **(b) Use additional text embeddings at inference time**. All experiments were performed with the same architecture [28], but with different text embeddings: gpt2-xl (first bullet), gpt2-xl-F (second bullet), the concatenation of gpt2-xl and gpt2-xl-F (third bullet), the mean of gpt2-xl and gpt2-xl-F (fourth bullet) and using TEACHTEXT (last bullet). By using multiple text embeddings at test time, which introduces an overhead, a boost in performance is obtained. However, by using TEACHTEXT there is no additional overhead at test time and the performance is superior.

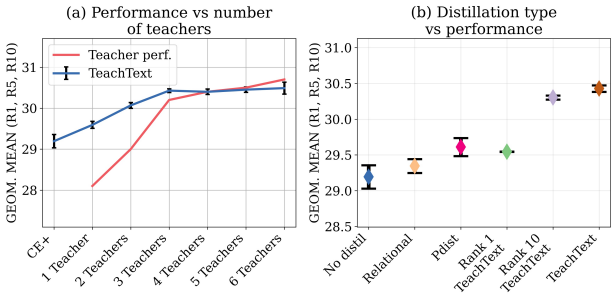

Figure 6. **(a) Teacher study.** We show the influence of learning from different number of teachers on the MSR-VTT dataset (all students share the same CE+ model, y-axes scaled for clarity). The teachers were added in the following order: gpt2-xl, w2v, gpt2-xl-F, mt_grovle, openai-gpt, gpt2-large. The performance of the combined teachers grows as more teachers are added, however it reaches a plateau after the first 3 teachers. The trend is similar for student performance. **(b) Distillation type.** Presenting various alternatives for distilling the information from the teacher: relational distillation [37] which preserves intra-text and intra-video relationships, pairwise distance distillation (*Pdist* - adapting [37] for cross modal relationships), ranking distillation inspired by [48] at Rank 1 and Rank 10 and TEACHTEXT. The first bullet represents the student without distillation.

Teacher variation. The teacher models share the same architecture with the student, but use a different text embedding. We next conduct an ablation on the influence of the number of used teachers. We observe in Fig. 6a that performance increases with the addition of more teachers. Since the combined performance of the teachers after adding more than 3 remains about the same, we do not obtain a further improvement. Thus, for our final experiments presented in Sec. 5.4 we use a combination of three teachers, having the following text embeddings: w2v [34], gpt2-xl [41] and gpt2-xl-F (gpt2-xl finetuned on the captions from the target dataset). A study of how each individual text embedding affects the final performance can be found in the Suppl. Mat. section *Teacher study*, where we observe that even when us-

Model	MSRVTT		MSRVTT 1k-A		MSVD		DiDeMo		LSMDC		ActivityNet	
	Base	TeachText	Base	TeachText	Base	TeachText	Base	TeachText	Base	TeachText	Base	TeachText
MoEE	$24.4_{\pm 0.1}$	$\mathbf{25.8}_{\pm 0.1}$	$41.6_{\pm 0.4}$	$\mathbf{43.4}_{\pm 0.6}$	$41.8_{\pm 0.3}$	$\mathbf{43.2}_{\pm 0.5}$	$33.2_{\pm 1.4}$	$\mathbf{40.2}_{\pm 0.7}$	$23.8_{\pm 0.4}$	$\mathbf{26.0}_{\pm 0.5}$	$40.1_{\pm 0.3}$	$\mathbf{45.2}_{\pm 0.1}$
CE	$24.4_{\pm 0.1}$	$\mathbf{25.9}_{\pm 0.1}$	$42.0_{\pm 0.8}$	$\mathbf{43.8}_{\pm 0.3}$	$42.3_{\pm 0.6}$	$\mathbf{42.6}_{\pm 0.4}$	$34.2_{\pm 0.4}$	$\mathbf{39.5}_{\pm 0.5}$	$23.7_{\pm 0.3}$	$\mathbf{25.5}_{\pm 0.5}$	$40.4_{\pm 0.3}$	$\mathbf{45.0}_{\pm 0.6}$
MMT	-	-	$44.7_{\pm 0.4}$	$\mathbf{45.6}_{\pm 0.7}$	-	-	-	-	$24.6_{\pm 0.7}$	$\mathbf{25.9}_{\pm 0.6}$	$44.0_{\pm 0.4}$	$\mathbf{47.9}_{\pm 0.4}$
CE+	$29.2_{\pm 0.2}$	$\mathbf{30.4}_{\pm 0.0}$	$50.3_{\pm 0.2}$	$\mathbf{50.9}_{\pm 0.4}$	$46.5_{\pm 1.0}$	$\mathbf{46.6}_{\pm 0.5}$	$35.8_{\pm 0.4}$	$\mathbf{40.4}_{\pm 0.4}$	$28.1_{\pm 0.3}$	$\mathbf{30.7}_{\pm 0.3}$	$39.7_{\pm 0.0}$	$\mathbf{46.3}_{\pm 0.2}$

Table 1. **Method generality**. Retrieval performance (geometric mean of R1-R5-R10) on various datasets when applying TeachText on top of different base models: MoEE[32], CE[28], MMT[18] (on available datasets) and CE+. We present in bold cases where TeachText brings an improvement over the base architecture. We observe that our method improves the performance for all underlying base models and on all datasets.

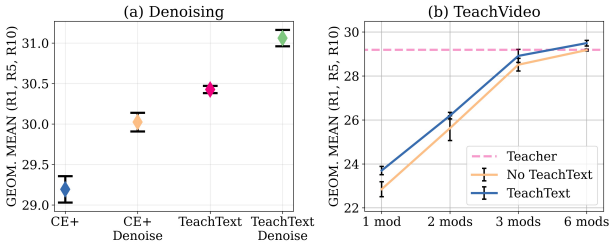

Figure 7. **(a) Denoising.** We present the effect of denoising on retrieval performance on MSR-VTT (y-axes scaled for clarity). Some of the captions available in datasets with multiple captions per video may be noisy and actively harm the training process. We estimate the degree of noise present in a caption by looking at the teacher rank and drop the caption if necessary. We observe the effectiveness of denoising when applied in isolation (*CE+* vs *CE+ Denoise*) and in conjunction with the full TeachText method. **(b) TeachVideo - Extension to video side modalities.** We observe that our method can be effective in taking advantage of the additional information brought by using multiple video side modalities, without incurring computational overhead at test time.

ing a teacher with lower performance (w2v), the student has a significant boost in performance.

Distillation ablation. We compare the proposed learning of the similarity matrix with other distillation alternatives. As seen in Fig.6b, our proposed approach is effective in capturing the relationships between video and text. We first provide comparisons between TeachText and several possible instantiations of relational distillation [37]. Indeed, given the highly general nature of [37], TeachText can be interpreted within this framework as a particular relational configuration that employs cross-modal distillation through batches of similarity matrices. Since the original work of [37] considered single-modality applications, we explore two variations of [37] as baselines for the text-video retrieval task. The first one (Relational), preserves the same intra-text and intra-video relationships independently. We use the same cost function as in [37] and enforce it on both video and text embeddings. The second approach (Pdist), uses the cross modal pairwise distances as a relation measure between text and video as opposed to the similarity matrix. While these methods indeed bring a gain, we observe that TeachText is more effective.

We also provide a baseline inspired by the work of [48] which highlights the importance of looking only at the top K predictions given by the teacher. To do so, we enforce the same similarities using TeachText only for the top K ranks given by the teacher rather than for the whole mini-batch. We show the performance for K=1 and K=10 (*Rank 1* and *Rank 10* presented in Fig.6b). Restricting to only top K predictions when distilling the similarity matrix results in a slight drop in performance.

Method generality. To demonstrate the generality of TeachText, we test it against three state of the art methods [18, 28, 32] in addition to the proposed CE+ baseline. In Tab. 1 we observe a consistent gain in performance, independent of the base architecture. Moreover, a gain is achieved across all the datasets that we tested, having over 5% absolute gain on DiDeMo and ActivityNet datasets for MoEE, CE and CE+ models. Note that for MMT [18] we report results on the datasets included in the public implementation provided by the authors[2].

Method application – Denoising. One immediate application of our method is data denoising. Existing real-world text-video datasets for the retrieval task suffer from label noise which can harm training. More concretely, in crowd-sourced datasets such as MSR-VTT there are some captions that are highly ambiguous/generic (e.g "A tutorial is presented", "Clip showing different colours", "A man is writing") and can describe multiple videos from the dataset. We therefore propose to use TeachText teachers to filter out such cases. For this scenario, we simply remove low-ranked predictions given by teachers and re-train the student using only the new samples. Specifically, we remove all sentences for which the correct video is not ranked in top 40 from the training set. This method is best-suited for datasets where multiple captions per video are available, ensuring that we can remove noisy captions without removing the video itself from training. Following this, we apply the denoising on MSR-VTT and MSVD datasets with the CE+ model. As seen in Fig. 7a, this can be an effective way of further improving the results. Please note, denoising is not used in any other ablations.

TeachVideo – Extension to video modalities. While the focus of this work is the use of multiple text embeddings, it is natural to consider whether this approach can be extended to the video encoder modalities. Thus, we introduce the TeachVideo algorithm which follows the same

[2] https://github.com/gabeur/mmt

Model	$R@1 \uparrow$	$R@5 \uparrow$	$R@10 \uparrow$	$MdR \downarrow$
Dual[16]	7.7	22.0	31.8	32.0
HGR[12]	9.2	26.2	36.5	24.0
MoEE[32][3]	$11.1_{\pm 0.1}$	$30.7_{\pm 0.1}$	$42.9_{\pm 0.1}$	$15.0_{\pm 0.0}$
CE[28][3]	$11.0_{\pm 0.0}$	$30.8_{\pm 0.1}$	$43.3_{\pm 0.3}$	$15.0_{\pm 0.0}$
TT-CE	$11.8_{\pm 0.1}$	$32.7_{\pm 0.1}$	$45.3_{\pm 0.1}$	$13.0_{\pm 0.0}$
TT-CE+	$\mathbf{15.0}_{\pm 0.1}$	$\mathbf{38.5}_{\pm 0.1}$	$\mathbf{51.7}_{\pm 0.1}$	$\mathbf{10.0}_{\pm 0.0}$

Table 2. **MSR-VTT full split: Comparison to state of the art.**

Model	$R@1 \uparrow$	$R@5 \uparrow$	$R@10 \uparrow$	$MdR \downarrow$
MoEE[32][3]	$21.6_{\pm 1.0}$	$50.8_{\pm 1.1}$	$65.6_{\pm 0.7}$	$5.3_{\pm 0.6}$
CE[28][3]	$21.7_{\pm 1.3}$	$51.8_{\pm 0.5}$	$65.7_{\pm 0.6}$	$5.0_{\pm 0.0}$
MMT[18]	$24.6_{\pm 0.4}$	$54.0_{\pm 0.2}$	$67.1_{\pm 0.5}$	$4.0_{\pm 0.0}$
SSB[39]	27.4	56.3	67.7	**3.0**
TT-MMT	$24.8_{\pm 0.2}$	$55.9_{\pm 0.7}$	$68.5_{\pm 1.0}$	$4.3_{\pm 0.5}$
TT-CE+	$\mathbf{29.6}_{\pm 0.3}$	$\mathbf{61.6}_{\pm 0.5}$	$\mathbf{74.2}_{\pm 0.3}$	$\mathbf{3.0}_{\pm 0.0}$

Table 3. **MSR-VTT 1k-A split[57]: Comparison with others.**

Model	$R@1 \uparrow$	$R@5 \uparrow$	$R@10 \uparrow$	$MdR \downarrow$
VSE++[17]	15.4	39.6	53.0	9.0
M-Cues[35]	20.3	47.8	61.1	6.0
MoEE[32][3]	$21.1_{\pm 0.2}$	$52.0_{\pm 0.7}$	$66.7_{\pm 0.2}$	$5.0_{\pm 0.0}$
CE[28][3]	$21.5_{\pm 0.5}$	$52.3_{\pm 0.8}$	$67.5_{\pm 0.7}$	$5.0_{\pm 0.0}$
TT-CE	$22.1_{\pm 0.4}$	$52.2_{\pm 0.5}$	$67.2_{\pm 0.6}$	$5.0_{\pm 0.0}$
TT-CE+	$\mathbf{25.4}_{\pm 0.3}$	$\mathbf{56.9}_{\pm 0.4}$	$\mathbf{71.3}_{\pm 0.2}$	$\mathbf{4.0}_{\pm 0.0}$

Table 4. **MSVD: Comparison to state of the art methods**.

Model	$R@1 \uparrow$	$R@5 \uparrow$	$R@10 \uparrow$	$MdR \downarrow$
S2VT[51]	11.9	33.6	–	13.0
FSE[60]	$13.9_{\pm 0.7}$	$36.0_{\pm 0.8}$	–	$11.0_{\pm 0.0}$
MoEE[32][3]	$16.1_{\pm 1.0}$	$41.2_{\pm 1.6}$	$55.2_{\pm 1.6}$	$8.3_{\pm 0.5}$
CE[28][3]	$17.1_{\pm 0.9}$	$41.9_{\pm 0.2}$	$56.0_{\pm 0.5}$	$8.0_{\pm 0.0}$
TT-CE	$21.0_{\pm 0.6}$	$47.5_{\pm 0.9}$	$61.9_{\pm 0.5}$	$\mathbf{6.0}_{\pm 0.0}$
TT-CE+	$\mathbf{21.6}_{\pm 0.7}$	$\mathbf{48.6}_{\pm 0.4}$	$\mathbf{62.9}_{\pm 0.6}$	$\mathbf{6.0}_{\pm 0.0}$

Table 5. **DiDeMo: Comparison to state of the art methods**.

Model	$R@1 \uparrow$	$R@5 \uparrow$	$R@10 \uparrow$	$MdR \downarrow$
JSFus[57]	9.1	21.2	34.1	36.0
MoEE[32][3]	$12.1_{\pm 0.7}$	$29.4_{\pm 0.8}$	$37.7_{\pm 0.2}$	$23.2_{\pm 0.8}$
CE[28][3]	$12.4_{\pm 0.7}$	$28.5_{\pm 0.8}$	$37.9_{\pm 0.6}$	$21.7_{\pm 0.6}$
MMT[18]	$13.2_{\pm 0.4}$	$29.2_{\pm 0.8}$	$38.8_{\pm 0.9}$	$21.0_{\pm 1.4}$
TT-MMT	$13.6_{\pm 0.5}$	$31.2_{\pm 0.4}$	$40.8_{\pm 0.5}$	$17.7_{\pm 0.5}$
TT-CE+	$\mathbf{17.2}_{\pm 0.4}$	$\mathbf{36.5}_{\pm 0.6}$	$\mathbf{46.3}_{\pm 0.3}$	$\mathbf{13.7}_{\pm 0.5}$

Table 6. **LSMDC: Comparison to state of the art methods**.

setup as the original TEACHTEXT, but now the teacher has access to multiple video modalities instead of multiple text modalities. In this study, all students and all teachers use the same text embedding, so we can assess the gains due to TEACHVIDEO. By employing TEACHVIDEO we retain the computational advantage of requiring fewer video modalities during inference. As it can be seen from our experiments presented in Fig. 7b, the method is effective and brings a boost over the original student. We believe this extension may be useful in scenarios in which limited computational resources are available during inference.

Qualitative examples and other ablation studies are presented in Suppl. Mat.

[3]Please note that the numbers reported are higher than in the original paper due to compression artefacts correction.

Model	$R@1 \uparrow$	$R@5 \uparrow$	$R@50 \uparrow$	$MdR \downarrow$
MoEE[32][3]	$19.7_{\pm 0.3}$	$50.0_{\pm 0.5}$	$92.0_{\pm 0.2}$	$5.3_{\pm 0.5}$
CE[28][3]	$19.9_{\pm 0.3}$	$50.1_{\pm 0.7}$	$92.2_{\pm 0.6}$	$5.3_{\pm 0.5}$
HSE[59]	20.5	49.3	–	–
MMT[18]	$22.7_{\pm 0.2}$	$54.2_{\pm 1.0}$	$93.2_{\pm 0.4}$	$5.0_{\pm 0.0}$
SSB[39]	**26.8**	58.1	93.5	**3.0**
TT-MMT	$25.0_{\pm 0.3}$	$\mathbf{58.7}_{\pm 0.4}$	$95.6_{\pm 0.2}$	$4.0_{\pm 0.0}$
TT-CE+	$23.5_{\pm 0.2}$	$57.2_{\pm 0.5}$	$\mathbf{96.1}_{\pm 0.1}$	$4.0_{\pm 0.0}$

Table 7. **ActivityNet: Comparison to state of the art methods**.

Model	$R@1 \uparrow$	$R@5 \uparrow$	$R@10 \uparrow$	$MdR \downarrow$
VSE[23]	28.0	64.3	76.9	3.0
Dual[16]	31.1	67.4	78.9	3.0
VSE++[17]	33.7	70.1	81.0	2.0
HGR[12]	35.1	73.5	83.5	2.0
SSB[39]	44.6	81.8	89.5	**1.0**
CE[28]	$47.9_{\pm 0.1}$	$84.2_{\pm 0.1}$	$91.3_{\pm 0.1}$	$2.0_{\pm 0.0}$
TT-CE	$49.7_{\pm 0.1}$	$85.6_{\pm 0.1}$	$92.4_{\pm 0.1}$	$2.0_{\pm 0.0}$
TT-CE+	$\mathbf{53.2}_{\pm 0.2}$	$\mathbf{87.4}_{\pm 0.1}$	$\mathbf{93.3}_{\pm 0.0}$	$\mathbf{1.0}_{\pm 0.0}$

Table 8. **VaTeX: Comparison to state of the art methods**.

Model	$R@1 \uparrow$	$R@5 \uparrow$	$R@10 \uparrow$	$MdR \downarrow$
MoEE[32]	$11.6_{\pm 1.3}$	$30.2_{\pm 3.0}$	$43.2_{\pm 3.1}$	$14.2_{\pm 1.6}$
CE[28]	$13.9_{\pm 0.8}$	$37.6_{\pm 1.2}$	$48.3_{\pm 1.4}$	$11.3_{\pm 0.6}$
TT-CE	$14.2_{\pm 1.4}$	$36.6_{\pm 2.0}$	$\mathbf{51.1}_{\pm 2.1}$	$\mathbf{9.7}_{\pm 1.2}$
TT-CE+	$\mathbf{14.4}_{\pm 0.5}$	$\mathbf{37.7}_{\pm 1.7}$	$50.9_{\pm 1.6}$	$9.8_{\pm 1.0}$

Table 9. **QuerYD: Comparison to state of the art methods**.

5.4. Comparison to prior work

As it can be seen in Tab.2,3,4,5,6,7,8,9 our approach is effective and achieves state of the art results on six datasets. All methods are trained for the retrieval task using only the samples from the target datasets. In order to be as fair as possible, we included the results of our TEACHTEXT (abbreviated TT) applied also to the best existing method for each dataset. So, the architecture and the used features are identical during inference (e.g. TT-CE has the same architecture and uses the same video and text embeddings as CE). We highlight in bold the best performing method.

6. Conclusion

In this paper, we present a novel algorithm TEACHTEXT for the text-video retrieval task. We use a teacher-student paradigm where a student learns to leverage the additional information given by one or multiple teachers, sharing the architecture, but each using a different pre-trained text embedding at input. In this way, we achieve state of the art results on six benchmarks. Finally, we present an application of our approach for denoising video retrieval datasets.

Acknowledgements. This work was supported by EPSRC Programme Grants Seebibyte EP/M013774/1 and VisualAI EP/T028572/1, and a gift from Adobe. M.L. was supported by UEFISCDI, under project EEA-RO-2018-0496. The authors would like to thank Gyungin Shin and Iulia Duta for assistance. S.A. would like to acknowledge the support of Z. Novak and S. Carlson in enabling his contribution.

References

[1] 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.

[2] Yusuf Aytar, Mubarak Shah, and Jiebo Luo. Utilizing semantic word similarity measures for video retrieval. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, 2008.

[3] Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman. Frozen in time: A joint video and image encoder for end-to-end retrieval. *arXiv preprint arXiv:2104.00650*, 2021.

[4] Oren Boiman and Michal Irani. Detecting irregularities in images and in video. *International journal of computer vision*, 74(1):17–31, 2007.

[5] Piotr Bojanowski, Rémi Lajugie, Edouard Grave, Francis Bach, Ivan Laptev, Jean Ponce, and Cordelia Schmid. Weakly-supervised alignment of video with text. In *Proceedings of the IEEE international conference on computer vision*, pages 4462–4470, 2015.

[6] Leo Breiman and Nong Shang. Born again trees. *University of California, Berkeley, Berkeley, CA, Technical Report*, 1:2, 1996.

[7] Cristian Buciluǎ, Rich Caruana, and Alexandru Niculescu-Mizil. Model compression. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 535–541, 2006.

[8] Andrea Burns, Reuben Tan, Kate Saenko, Stan Sclaroff, and Bryan A Plummer. Language features matter: Effective language representations for vision-language tasks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 7474–7483, 2019.

[9] Fabian Caba Heilbron, Victor Escorcia, Bernard Ghanem, and Juan Carlos Niebles. Activitynet: A large-scale video benchmark for human activity understanding. In *Proceedings of the ieee conference on computer vision and pattern recognition*, pages 961–970, 2015.

[10] Yue Cao, Mingsheng Long, Jianmin Wang, and Shichen Liu. Deep visual-semantic quantization for efficient image retrieval. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1328–1337, 2017.

[11] David L 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-Volume 1*, pages 190–200. Association for Computational Linguistics, 2011.

[12] Shizhe Chen, Yida Zhao, Qin Jin, and Qi Wu. Fine-grained video-text retrieval with hierarchical graph reasoning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10638–10647, 2020.

[13] Ondrej Chum, James Philbin, Michael Isard, and Andrew Zisserman. Scalable near identical image and shot detection. In *CIVR*, 2007.

[14] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, 2019.

[15] Jianfeng Dong, Xirong Li, and Cees GM Snoek. Word2visualvec: Image and video to sentence matching by visual feature prediction. *arXiv preprint arXiv:1604.06838*, 2016.

[16] Jianfeng Dong, Xirong Li, Chaoxi Xu, Shouling Ji, and Xun Wang. Dual dense encoding for zero-example video retrieval. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

[17] Fartash Faghri, David J Fleet, Jamie Ryan Kiros, and Sanja Fidler. Vse++: Improving visual-semantic embeddings with hard negatives. *arXiv preprint arXiv:1707.05612*, 2017.

[18] Valentin Gabeur, Chen Sun, Karteek Alahari, and Cordelia Schmid. Multi-modal transformer for video retrieval. *European Conference on Computer Vision*, 2020.

[19] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

[20] Zehao Huang and Naiyan Wang. Like what you like: Knowledge distill via neuron selectivity transfer. *arXiv preprint arXiv:1707.01219*, 2017.

[21] Yu-Gang Jiang, Chong-Wah Ngo, and Jun Yang. Towards optimal bag-of-features for object categorization and semantic video retrieval. In *Proceedings of the 6th ACM international conference on Image and video retrieval*, pages 494–501, 2007.

[22] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[23] Ryan Kiros, Ruslan Salakhutdinov, and Richard S Zemel. Unifying visual-semantic embeddings with multimodal neural language models. *arXiv preprint arXiv:1411.2539*, 2014.

[24] Bruno Korbar, Fabio Petroni, Rohit Girdhar, and Lorenzo Torresani. Video understanding as machine translation. *arXiv preprint arXiv:2006.07203*, 2020.

[25] Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. Albert: A lite bert for self-supervised learning of language representations. *arXiv preprint arXiv:1909.11942*, 2019.

[26] Ivan Laptev and Patrick Pérez. Retrieving actions in movies. In *2007 IEEE 11th International Conference on Computer Vision*, pages 1–8. IEEE, 2007.

[27] Yong-Lu Li, Xinpeng Liu, Xiaoqian Wu, Yizhuo Li, and Cewu Lu. Hoi analysis: Integrating and decomposing human-object interaction. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 5011–5022. Curran Associates, Inc., 2020.

[28] Yang Liu, Samuel Albanie, Arsha Nagrani, and Andrew Zisserman. Use what you have: Video retrieval using representations from collaborative experts. *arXiv preprint arXiv:1907.13487*, 2019.

[29] 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. *arXiv preprint arXiv:1907.11692*,

2019.

[30] David Lopez-Paz, Léon Bottou, Bernhard Schölkopf, and Vladimir Vapnik. Unifying distillation and privileged information. *ICLR*, 2016.

[31] 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. *arXiv preprint arXiv:1912.06430*, 2019.

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

[33] 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 International Conference on Computer Vision*, pages 2630–2640, 2019.

[34] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

[35] Niluthpol Chowdhury Mithun, Juncheng Li, Florian Metze, and Amit K Roy-Chowdhury. Learning joint embedding with multimodal cues for cross-modal video-text retrieval. In *Proceedings of the 2018 ACM on International Conference on Multimedia Retrieval*, pages 19–27, 2018.

[36] Andreea-Maria Oncescu, Joao F. Henriques, Yang Liu, Andrew Zisserman Zisserman, and Samuel Albanie. Queryd: a video dataset with high-quality textual and audio narrations. *arXiv preprint arXiv:2011.11071*, 2020.

[37] Wonpyo Park, Dongju Kim, Yan Lu, and Minsu Cho. Relational knowledge distillation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3967–3976, 2019.

[38] 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. In *Advances in neural information processing systems*, pages 8026–8037, 2019.

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

[40] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. Improving language understanding by generative pre-training. *URL https://s3-us-west-2. amazonaws. com/openai-assets/researchcovers/languageunsupervised/language understanding paper. pdf*, 2018.

[41] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. *preprint*, 1(8):9, 2019.

[42] Anna Rohrbach, Atousa Torabi, Marcus Rohrbach, Niket Tandon, Christopher Pal, Hugo Larochelle, Aaron Courville, and Bernt Schiele. Movie description. *International Journal of Computer Vision*, 123(1):94–120, 2017.

[43] Adriana Romero, Nicolas Ballas, Samira Ebrahimi Kahou, Antoine Chassang, Carlo Gatta, and Yoshua Bengio. Fitnets: Hints for thin deep nets. *arXiv preprint arXiv:1412.6550*, 2014.

[44] Lifeng Shang, Linjun Yang, Fei Wang, Kwok-Ping Chan, and Xian-Sheng Hua. Real-time large scale near-duplicate web video retrieval. In *Proceedings of the 18th ACM international conference on Multimedia*, pages 531–540, 2010.

[45] Josef Sivic and Andrew Zisserman. Video google: A text retrieval approach to object matching in videos. In *ICCV*, page 1470. IEEE, 2003.

[46] Cees GM Snoek and Marcel Worring. Multimodal video indexing: A review of the state-of-the-art. *Multimedia tools and applications*, 25(1):5–35, 2005.

[47] Richard Socher, Andrej Karpathy, Quoc V Le, Christopher D Manning, and Andrew Y Ng. Grounded compositional semantics for finding and describing images with sentences. *Transactions of the Association for Computational Linguistics*, 2:207–218, 2014.

[48] Jiaxi Tang and Ke Wang. Ranking distillation: Learning compact ranking models with high performance for recommender system. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 2289–2298, 2018.

[49] Vladimir Vapnik and Rauf Izmailov. Learning using privileged information: similarity control and knowledge transfer. *J. Mach. Learn. Res.*, 16(1):2023–2049, 2015.

[50] Vladimir Vapnik and Akshay Vashist. A new learning paradigm: Learning using privileged information. *Neural networks*, 22(5-6):544–557, 2009.

[51] Subhashini Venugopalan, Huijuan Xu, Jeff Donahue, Marcus Rohrbach, Raymond Mooney, and Kate Saenko. Translating videos to natural language using deep recurrent neural networks. *arXiv preprint arXiv:1412.4729*, 2014.

[52] Qingzhong Wang and Antoni B Chan. Describing like humans: on diversity in image captioning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4195–4203, 2019.

[53] Xin Wang, Jiawei Wu, Junkun Chen, Lei Li, Yuan-Fang Wang, and William Yang Wang. Vatex: A large-scale, high-quality multilingual dataset for video-and-language research. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4581–4591, 2019.

[54] Michael Wray, Diane Larlus, Gabriela Csurka, and Dima Damen. Fine-grained action retrieval through multiple parts-of-speech embeddings. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 450–459, 2019.

[55] 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.

[56] Ran Xu, Caiming Xiong, Wei Chen, and Jason J Corso. Jointly modeling deep video and compositional text to bridge vision and language in a unified framework. In *Twenty-Ninth AAAI Conference on Artificial Intelligence*, 2015.

[57] 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 (ECCV)*, pages 471–487, 2018.

[58] Sergey Zagoruyko and Nikos Komodakis. Paying more attention to attention: Improving the performance of convolu-

tional neural networks via attention transfer. *arXiv preprint arXiv:1612.03928*, 2016.

[59] 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 (ECCV)*, pages 374–390, 2018.

[60] Michael Zhang, James Lucas, Jimmy Ba, and Geoffrey E Hinton. Lookahead optimizer: k steps forward, 1 step back. In *Advances in Neural Information Processing Systems*, pages 9597–9608, 2019.