This CVPR paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

Text Is MASS: Modeling as Stochastic Embedding for Text-Video Retrieval

Jiamian Wang[1], Guohao Sun[1], Pichao Wang[2]*, Dongfang Liu[1]†,
Sohail Dianat[1], Majid Rabbani[1], Raghuveer Rao[3], Zhiqiang Tao[1]†
[1]Rochester Institute of Technology, [2]Amazon Prime Video, [3]Army Research Laboratory

Abstract

The increasing prevalence of video clips has sparked growing interest in text-video retrieval. Recent advances focus on establishing a joint embedding space for text and video, relying on consistent embedding representations to compute similarity. However, the text content in existing datasets is generally short and concise, making it hard to fully describe the redundant semantics of a video. Correspondingly, a single text embedding may be less expressive to capture the video embedding and empower the retrieval. In this study, we propose a new stochastic text modeling method T-MASS, i.e., text is modeled as a stochastic embedding, to enrich text embedding with a flexible and resilient semantic range, yielding a text mass. To be specific, we introduce a similarity-aware radius module to adapt the scale of the text mass upon the given text-video pairs. Plus, we design and develop a support text regularization to further control the text mass during the training. The inference pipeline is also tailored to fully exploit the text mass for accurate retrieval. Empirical evidence suggests that T-MASS not only effectively attracts relevant text-video pairs while distancing irrelevant ones, but also enables the determination of precise text embeddings for relevant pairs. Our experimental results show a substantial improvement of T-MASS over baseline ($3\% \sim 6.3\%$ by R@1). Also, T-MASS achieves state-of-the-art performance on five benchmark datasets, including MSRVTT, LSMDC, DiDeMo, VATEX, and Charades. Code and models are available here.

1. Introduction

Text-video retrieval is to find the most semantically relevant video clip (text) from a candidate pool referring to the text (video clip) query [3, 5, 7, 39, 43, 57]. Performing an accurate retrieval is non-trivial due to the divergent characteristics of video and text: videos tend to offer redundant semantic clues [13, 17, 60], inevitably posing challenges for the feature extraction, while the text, commonly appearing as short captions, subtitles, and even hashtags, seems to be semantically limited by comparison to videos [32].

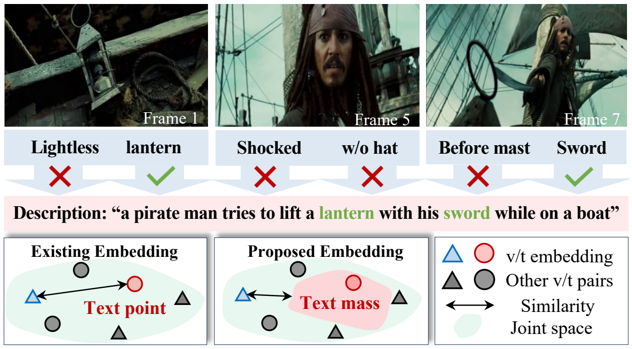

Figure 1. Text inside a relevant video is hard to fully describe the redundant semantics of the video. Correspondingly, single text embedding may be less expressive to handle the video information in joint space. We propose a new embedding of text mass with a resilient semantic range, to better capture rich video clues.

Recognizing the nature of video and text, some prior works [39, 57] adapt powerful vision-language models (e.g., CLIP [44]) to the multimodal domain [4] of text and video. Others learn enhanced video representation through video-text interaction [17, 32, 48] or temporal modeling [3, 36]. Besides, bridging video and text at a fine granularity also forms a promising direction [18, 24, 25, 49, 51, 61]. In summary, prevailing text-video retrieval methods are mainly dedicated to extracting accurate video or text embedding, such as text/video points, for retrieval.

Despite the success of video/text embedding methods, it is hard to learn a single text embedding to fully cover all the semantics and visual variations inside a video, since the text content is usually short and concise, which contains limited semantics compared with its paired relevant video (see Fig. 1 *top*). This fact exacerbates alignment difficulties, where text may not adequately express the richness of video information. Drawing inspiration, we provide a more flexible text modeling approach to capture rich video semantic clues, thus enabling a better alignment between video and text semantics. We introduce T-MASS, *i.e.*, **T**ext is

[*]The work does not relate to author's position at Amazon.
[†]Corresponding authors: Dongfang Liu (dongfang.liu@rit.edu) and Zhiqiang Tao (zhiqiang.tao@rit.edu)

Modeled As a StochStic embedding. Unlike existing methods, the proposed method no longer treats text as a single point in the embedding space, but projects it as a "mass" (see Fig. 1 *bottom*) to enable a resilient semantic range to account for the potential misalignment between video and text embeddings. A straightforward way to implement text mass can adopt the repremetrization [27] upon the deterministic text embedding given by CLIP. However, learning such a text mass imposes the following challenges.

First, it is non-trivial to determine the scale of the text mass. The underlying scale is text-dependent and can even be dynamic relative to different videos. To this end, we develop a similarity-aware radius module to enable a learnable scale adaptive to text-video pairs. Second, how to further regularize and shift the text mass in the joint embedding space is an open question. Without jointly processing the whole text mass, we find that solely performing contrastive learning between sampled stochastic text points and video points during training brings promising performance. Besides, we locate a support text vector upon the text mass, taking it as a proxy to simultaneously control the position and scale of the text mass relative to the query video. We also reformulate the inference process to fully exploit the text mass for more effective text-video retrieval. For each video candidate, we first sample a batch of stochastic text embedding for the query text and choose the closest one to the video embedding for the evaluation. Interestingly, we find that the proposed T-MASS not only bridges the relevant pairs and pushes the irrelevant ones (Fig. 5), compared with the single-point text representation, but also empowers a precise text semantics mapping (Fig. 3). We summarize the contributions of this work as follows.

- This work rethinks the design of text embedding for text-video retrieval. We propose T-MASS as a new stochastic modeling approach to enable expressive and flexible text embedding to better capture video clues and align text and video semantics in joint space.
- This work provides a representative design of similarity-aware radius network to encourage the text semantics alignment, facilitating a resilient and flexible text embedding that can adapt to the video variations.
- This work develops an effective learning strategy upon stochastic text embedding, specifically, a stochastic symmetric cross-entropy learning objective to learn an effective text mass. Besides, a support text vector as a regularization to further scale and shift the text mass.
- The proposed method improves the baseline by a large margin ($+3\% \sim 6.3\%$ at R@1), setting the new state-of-the-art on five benchmark datasets, including MSRVTT, LSMDC, DiDeMo, Charades, and VATEX. Extensive analyses find that T-MASS not only better distances irrelevant pairs and attracts relevant pairs, but also enables a more promising text semantics learning for relevant pairs.

2. Related Work

Text-video Retrieval. JSFusion [58] pioneered the exploration of hierarchical similarities between video and text using a convolutional decoder, establishing a benchmark for the task. Transformer [11, 47]-based methods [9, 12, 15, 16, 20, 22, 29] abstract multi-modal data clues via cross attention, resulting in significant performance gains. Recent advancements leverage CLIP [44] for the semantics extraction [17, 28, 39, 53, 54, 57, 60], *e.g.*, CLIP4Clip [39] discusses the transferability of pre-trained CLIP model to text-video retrieval. To solve the domain gap, CLIP-ViP [57] exploits the video post-pretraining, achieving state-of-the-art results. Alternatively, Cap4Video [53] harnesses the power of a pre-trained large model by introducing additional captions, bringing insight on fully taking advantage of augmented data. TEACHTEXT [8] empowers the retrieval by leveraging multiple text encoders. Besides, DiffusionRet [26] advances by integrating diffusion model into the text-video retrieval. Additionally, introducing additional modality, *e.g.*, audio [1, 21, 33, 35, 40], draws increasing attention. The proposed method opts to learn a expressive and powerful text embedding, achieving substantial improvements without post-pretrain the CLIP with additional video data. The proposed method can even outperform previous methods enhanced by post-processing techniques [5, 7].

Text and Video Representation Learning. This work builds upon CLIP [44] model, attributing to its promising semantics extraction. Based on CLIP, existing methods predominantly focus on enhanced video and text representations for retrieval [10, 14, 17–19, 23, 30, 34, 42, 59]. TS2-Net [36] models fine-grained temporal visual clues, showcasing promising performance. X-Pool [17] exploits text-conditioned feature fusion across frames, delivering more semantically similar embedding. PIDRo [18] and ProST [30] model the informative semantic clues of video and text in a fine-grained manner, achieving encouraging performances. UATVR [13] innovatively recognizes and models uncertainties in both modalities. In contrast to these methods representing text and video embedding with a common form, T-MASS implements a stochastic text embedding, jointly learning a text mass and a video point in embedding space. Notably, the proposed method achieves remarkable performance boost without requiring sophisticated designs on video feature extraction, such as temporal modeling [3, 31, 36], fine-grained alignment [6, 51, 52], etc.

3. Method

3.1. Preliminaries

We denote the text as t and the raw video clip as v. The task of text-video retrieval firstly involves learning embedding for text and video in a joint space, yielding $\mathbf{t}, \mathbf{v} \in \mathbb{R}^d$, where d represents the feature dimension. A similarity mea-

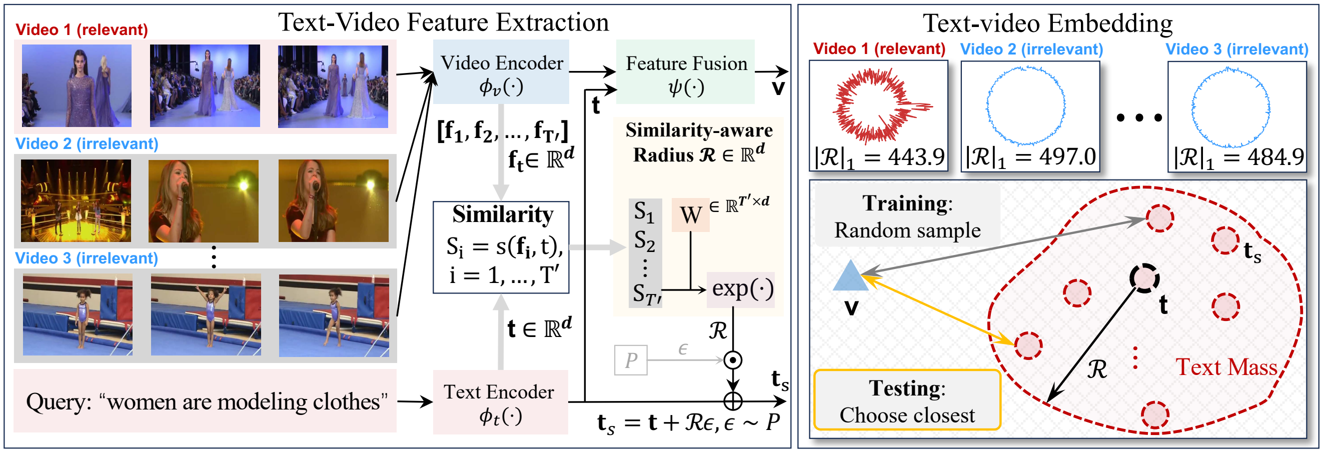
Text-Video Feature Extraction
Video 1 (relevant)
Video 2 (irrelevant)
Video 3 (irrelevant)
Query: "women are modeling clothes"
Video Encoder $\phi_v(\cdot)$
Feature Fusion $\psi(\cdot)$
$\mathbf{v}$
$\mathbf{t}$
$[\mathbf{f_1}, \mathbf{f_2}, ..., \mathbf{f_{T'}}]$
$\mathbf{f_t} \in \mathbb{R}^d$
Similarity
$S_i = s(\mathbf{f_i}, \mathbf{t})$, $i = 1, ..., T'$
Similarity-aware Radius $\mathcal{R} \in \mathbb{R}^d$
S_1
S_2
$S_{T'}$
$W \in \mathbb{R}^{T' \times d}$
$\exp(\cdot)$
$\mathcal{R}$
P
ϵ
$\mathbf{t} \in \mathbb{R}^d$
$\mathbf{t}_s$
Text Encoder $\phi_t(\cdot)$
$\mathbf{t}_s = \mathbf{t} + \mathcal{R}\epsilon, \epsilon \sim P$
Text-video Embedding
Video 1 (relevant)
Video 2 (irrelevant)
Video 3 (irrelevant)
$|\mathcal{R}|_1 = 443.9$
$|\mathcal{R}|_1 = 497.0$
$|\mathcal{R}|_1 = 484.9$
Training: Random sample
$\mathbf{v}$
$\mathbf{t}_s$
$\mathbf{t}$
$\mathcal{R}$
Text Mass
Testing: Choose closest

Figure 2. Illustration of the proposed text-video retrieval method T-MASS, which adopts dual-branch CLIP [44] (ϕ_v and ϕ_t) to extract frame features $[\mathbf{f}_1, ..., \mathbf{f}_{T'}]$ and text embedding $\mathbf{t}$. Then a feature fusion module ψ is employed to produce video embedding $\mathbf{v}$. We develop a similarity-aware module $\mathcal{R}$ to facilitate the reparameterization [27] of the stochastic text embedding $\mathbf{t}_s$, yielding a text mass in the joint space. During training, we compute the loss upon $\mathbf{v}$ and random sampled $\mathbf{t}_s$. During evaluation, we collect a group of $\mathbf{t}_s$ and select the one exhibiting the highest similarity with $\mathbf{v}$. We visualize the learned radius $\mathcal{R}$ for relevant/irrelevant pairs. More details are in Section 3.3.

suring function $s(\mathbf{t}, \mathbf{v})$, such as cosine similarity, is then employed to compute relevancy. Given a training dataset with K different text-video pairs, $\mathcal{D} = \{(t_k, v_k)\}_{k=1}^{k=K}$, a widely-used loss function for this task can be a symmetric cross entropy [41], which minimizes the distance of relevant text-video pairs while maximizes distance of irrelevant pairs. Both text-to-video ($t \rightarrow v$) and video-to-text ($v \rightarrow t$) are considered in this approach by

$$\begin{aligned} \mathcal{L}_{t\rightarrow v} &= -\frac{1}{N}\sum_{i=1}^{N} \log \frac{e^{s(\mathbf{t}_i, \mathbf{v}_i)\cdot\lambda}}{\sum_j e^{s(\mathbf{t}_i, \mathbf{v}_j)\cdot\lambda}}, \\ \mathcal{L}_{v\rightarrow t} &= -\frac{1}{N}\sum_{i=1}^{N} \log \frac{e^{s(\mathbf{t}_i, \mathbf{v}_i)\cdot\lambda}}{\sum_j e^{s(\mathbf{t}_j, \mathbf{v}_i)\cdot\lambda}}, \end{aligned} \tag{1}$$

where N is a collection of text-video pairs, typically representing the batch size, and λ is a learnable scaling factor. Text and video embedding, $\mathbf{t}_i$ and $\mathbf{v}_i/\mathbf{v}_j$, are produced by delicate feature extractors with learnable parameters. The overall loss function $\mathcal{L}_{\text{ce}}$ is

$$\mathcal{L}_{\text{ce}} = \frac{1}{2}(\mathcal{L}_{t\rightarrow v} + \mathcal{L}_{v\rightarrow t}). \tag{2}$$

The loss function reaches zero when all text-video pairs in a batch is entirely relevant, *i.e.*, $s(\mathbf{t}_i, \mathbf{v}_i) = 1$, and $s(\mathbf{t}_i, \mathbf{v}_j) = 0$, $i \neq j$, for all irrelevant pairs. This is non-trivial and highly depends on the quality of text and video embedding ($\mathbf{t}$ and $\mathbf{v}$). As shown in Fig. 1, in practice, even text-video pairs that are identified as "relevant" could be not entirely consistent – video v provides redundant clues, while text t may contain limited semantics. This poses challenges to the semantics extraction for both modalities.

3.2. Text-Video Representations

Feature Extraction. The extraction of multi-modal semantic embedding has gained much attention [15, 58]. Recent advancement of CLIP [44] in recent text-video retrieval methods [17, 57], and thus we primarily focus on CLIP-based methods in this work. Given a video comprising T frames, denoted as $v = [f_1, ..., f_T]$, the widely-used protocol is to sample T' frames and feed them into CLIP, producing T' different frame embedding $\mathbf{f}_i$, $i = 1, ..., T'$. Let ϕ_v and ϕ_t denote the CLIP's image and text encoders. The feature extraction is given by

$$\mathbf{f}_i = \phi_v(f_i), i = 1, ..., T'; \;\; \mathbf{t} = \phi_t(t), \tag{3}$$

where $\mathbf{f}_i \in \mathbb{R}^d$. Based on the frame embedding $[\mathbf{f}_1, ..., \mathbf{f}_{T'}]$, previous works develop various strategies to compute the final video embedding $\mathbf{v}$ for the similarity measurement

$$\mathbf{v} = \psi([\mathbf{f}_1, ..., \mathbf{f}_{T'}], \mathbf{t}), \tag{4}$$

where $\psi(\cdot)$ is the feature fusion module that abstracts the video semantics through frame-text interaction under different granularities, or relies on temporal modeling, etc. Despite the promising performance, it seems that finding a close alignment between $\mathbf{t}$ and $\mathbf{v}$ is still challenging since this requires an effective $\phi_v(\cdot)$ and $\psi(\cdot)$ for the video embedding learning, also calls for a powerful $\phi_t(\cdot)$ for text embedding determination. This motivates us to re-examine the text and video embedding as follows.

Motivation. Existing methods emphasize more on learning video embedding $\mathbf{v}$, including frame sampling protocol, feature extraction $\phi_v(\cdot)$, and fusion $\psi(\cdot)$ designs, but pay less attention to text embedding $\mathbf{t}$. As shown in Section 1 and Fig. 1, text t is hard to fully describe the semantics of a video v, which yields $\mathbf{t}$ with less expressiveness and semantic clues to align with $\mathbf{v}$ in joint space. We are motivated to enhance the text embedding with more resilience and flexibility. Specifically, rather than using a single point, text embedding can be associated with a specific semantics *range* that is resilient enough to incorporate (or be close to) the relevant video embedding. This leads us to introduce a new embedding called text mass, setting it apart from existing methodologies.

3.3. Proposed Method: T-MASS

Stochastic Text Modeling. In this work, we introduce T-MASS, *i.e.*, **T**ext is **M**odeled **A**s a **S**tocha**S**tic representation. In contrast to prevalent treatments, T-MASS projects text as a "mass" to encourage expressive and resilient representations, jointly learning upon text-video embedding with distinct forms. Fig. 2 provides the framework of the proposed T-MASS. Specifically, we adopt reparameterization [27] to enable stochastic gradient calculations during the training. Based on the text embedding $\mathbf{t}$ given in Eq. (3), we introduce stochastic text embedding $\mathbf{t}_s \in \mathbb{R}^d$ as

$$\mathbf{t}_s = \mathbf{t} + \mathcal{R} \cdot \epsilon, \epsilon \sim P, \tag{5}$$

where ϵ is an auxiliary variable sampled from a prior distribution, *e.g.*, $P = \mathcal{N}(\mathbf{0}, \mathbf{1})$. $\mathcal{R} \in \mathbb{R}^d$ models the scale of the text mass and defines its underlying "radius". Unlike previous embedding that aims to appropriately adjust the distance between two points, *i.e.*, $\mathbf{t}$ and $\mathbf{v}$, any point that falls into the text mass is considered as a valid representation corresponding to the content of text t and can be used for the similarity calculation, *e.g.*, $s(\mathbf{t}_s, \mathbf{v})$. By this means, existing powerful text encoders can be naturally adopted, and minimum adjustments in implementation is required.

Similarity-Aware Radius Modeling. Directly incorporating stochastic text embedding in Eq. (5) into the loss $\mathcal{L}_{\text{ce}}$ in Eq. (2) is challenging. On the one hand, an oversized text mass ($\mathcal{R}$ is too large) might improperly encompass (or improperly be close to) less relevant or even irrelevant video embedding points in the joint space, thus misleading the retrieval, On the other hand, too small text mass ($\mathcal{R} \to 0$) may lacks expressiveness to bridge the video. Thereby, it is non-trivial to manually determine an optimal value for $\mathcal{R}$; rather, the underlying radius $\mathcal{R}$ should adapt to different text-video pairs. We propose a similarity-aware radius module to learn proper text mass scaling by jointly taking text $\mathbf{t}$ and video frames $[\mathbf{f}_1, ..., \mathbf{f}_{T'}]$ before feature fusion as inputs.

The key idea is to first compute the cosine similarity of a text-video pair and leverage it as an indicator of the text-video relationship. For instance, if the text-video pair exhibits relevance, we expect a well-aligned text mass with a proper radius and position that potentially allows an accurate retrieval, as shown by the red curve in Fig. 3. Reversely, it is less likely to learn a meaningful mass when they are irrelevant, *e.g.*, imprecise text mass (blue curves in Fig. 3).

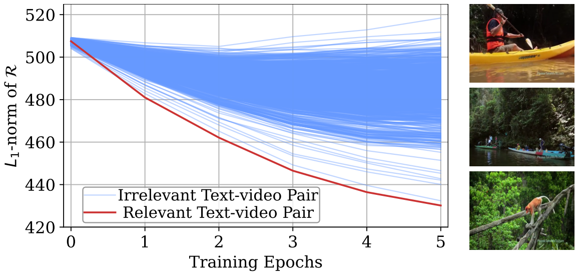

Figure 3. Dynamics of $\mathcal{R}$. We plot $|\mathcal{R}|_1$ for a relevant t-v pair (130-th in MSRVTT-1K, video on the *right*) and the query text with 999 irrelevant videos. T-MASS learns a precise text semantics for the relevant pair (smallest $|\mathcal{R}|_1$). This is typically observed on correctly retrieved pairs. More examples are in supplementary.

Given the embedding $\mathbf{t}$ and frame embedding $[\mathbf{f}_1, ..., \mathbf{f}_{T'}]$ by Eq. (3), we compute the text-video similarity as

$$S_i = s(\mathbf{t}, \mathbf{f}_i), i = 1, ..., T', \tag{6}$$

based on which we propose a learnable scalar θ to compute the radius $\mathcal{R} = \exp(\frac{\theta}{T'} \sum_{i=1}^{T'} S_i)$ where θ is broadcast to fit the dimension d. We use an exponential function to scale the radius further. We observe that such an implementation has brought an encouraging performance boost, compared with the unlearnable strategy, *i.e.*, defining $\mathcal{R}$ purely upon similarities values given by Eq. (6). Besides, solely using a scalar to adjust the radius in a high-dimensional space might be less flexible. As shown in Fig. 2, we also explore a linear layer to compute $\mathcal{R}$ by

$$\mathcal{R} = \exp(\mathbf{SW}), \mathbf{S} = [S_1, ..., S_{T'}], \tag{7}$$

where $\mathbf{W} \in \mathbb{R}^{T' \times d}$ denotes the learnable weights in the linear layer. The resulting $\mathcal{R}$ will take effect in the stochastic text embedding calculation as shown in Eq. (5). In Section 4.3, we provide the corresponding discussion toward the design principles of $\mathcal{R}$. See more details in Table 5.

Learning Text Mass in Joint Space. The original loss function in Eq. (1) between $\mathbf{t}$ and $\mathbf{v}$ may only take effect on shifting the text mass without controlling its scale. Since the text mass is implemented by stochastic text embedding, we randomly sample a stochastic text embedding $\mathbf{t}_s$ and use it to replace the text embedding $\mathbf{t}$ in Eq. (1) during training, so that different points in text mass participate into the learning. Distinguished from the original symmetric cross-entropy loss $\mathcal{L}_{\text{ce}}$, we denote this stochastic loss $\mathcal{L}_s$. The overall loss function becomes $\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{ce}} + \mathcal{L}_s$. We show that such a learning schedule brings notable benefits (*e.g.*, $> 1.5\%+$ at R@1). Moreover, we find that specifically regularizing over $\mathbf{t}$ (*i.e.*, using $\mathcal{L}_{\text{ce}}$) during the learning is unnecessary and can even be harmful. As, compared to $\mathbf{t}_s$, $\mathbf{t}$ cannot reflect the context and position of the text mass, thus focusing on $\mathbf{t}$ can lead to a biased text mass learning. In addition, given that the text mass presents as an irregular volume in a complex and high-dimensional embedding space, learning a limited amount of $\mathbf{t}_s$ points seems insufficient to regularize the whole text mass. A straightforward solution is to introduce KL-divergence to further control the scale. However, it is challenging to determine an optimal prior.

We propose to identify a support text embedding vector located at the border of the text mass as a proxy to help adjust the scale and the shift of the text mass, termed as support text vector $\mathbf{t}_{\text{sup}}$. As shown in Fig. 4, $\mathbf{t}_{\text{sup}}$ serves as a stochastic text embedding sample locating along the direction from $\mathbf{v}$ to $\mathbf{t}$ and being placed at the surface of the text mass. Therefore, pulling $\mathbf{v}$ and $\mathbf{t}_{\text{sup}}$ together or pushing them away to a large extent can help manipulate the text mass. We compute $\mathbf{t}_{\text{sup}}$ based on $\mathbf{t}$ given in Eq. (3), video embedding $\mathbf{v}$ upon Eq. (4), and the radius modeling $\mathcal{R}$ as

$$\mathbf{t}_{\text{sup}} = \mathbf{t} + \frac{\mathbf{v}-\mathbf{t}}{|\mathbf{v}-\mathbf{t}|}\mathcal{R}, \tag{8}$$

based on which we introduce another contrastive loss term with the same formulation as Eq. (2) but only exchange $\mathbf{t}$ with $\mathbf{t}_{\text{sup}}$. We denote this regularization as $\mathcal{L}_{\text{sup}}$. During training, we not only sample a stochastic text embedding to compute the contrastive loss with the video, but also constantly pay attention to the support text vector. Our experiment in Section 4.3 shows that such a regularization brings a remarkable performance boost. The resulting loss function of the proposed method is given by

$$\mathcal{L}_{\text{total}} = \mathcal{L}_s + \alpha\mathcal{L}_{\text{sup}}, \tag{9}$$

where α is the support text regularization weight. In Section 4.3 and Table 6a, we provide a complete comparison of different learning strategies. Besides, the proposed learning strategy encourages a better alignment for relevant/irrelevant text-video pairs[1]. See Fig. 5 for more details.

Inference pipeline. Building upon the proposed stochastic text representation, we modify the inference pipeline to take advantage of text mass. For any given text-video pairs $\{t, v\}$, we first extract text and frame features, $\mathbf{t}$ and $[\mathbf{f}_1, ..., \mathbf{f}_{T'}]$ using Eq. (3). Subsequently, we conduct M times stochastic sampling upon Eq. (5), producing $\{\mathbf{t}_s^1, ..., \mathbf{t}_s^M\}$. We then select an optimal text embedding that gives the highest similarity with the video by

$$\widehat{\mathbf{t}}_s = \arg\max_{\mathbf{t}_s}\ s(\mathbf{t}_s^i, \mathbf{v}), \quad i = 1, ..., M, \tag{10}$$

where $\mathbf{v}$ is computed by the feature fusion module $\psi(\cdot)$ according to Eq. (4). $\widehat{\mathbf{t}}_s$ is the final text embedding selected from the text mass for the metric computation. This strategy ensures that text embedding linked to the input text t is no longer fixed and adaptive to videos, which holds benefits for retrieval by exploring more possibilities of the text embedding (especially, the ones that are closer to the video than the original $\mathbf{t}$). Note that this strategy is applicable for both text-to-video and video-to-text retrievals.

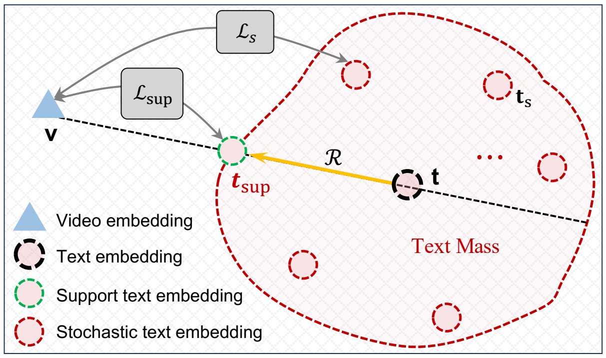

Figure 4. Support text regularization. Besides computing the loss between the video embedding $\mathbf{v}$ and stochastic text embedding $\mathbf{t}_s$, we identify a support text embedding locating along the direction from $\mathbf{v}$ to $\mathbf{t}$ and being placed at the surface of the text mass, which serves as a proxy to enable text mass shifting and scaling.

In summary, we introduce T-MASS, a stochastic text modeling method for for text-video retrieval. Diverging from existing methods, T-MASS advances the retrieval by empowering text embedding with more expressiveness and flexibility. Besides the encouraging performance, T-MASS enables a better text-video alignment and text semantics adaptation. See detailed illustrations and analysis below.

4. Experiment

4.1. Experimental Settings

Datasets and Metrics. We adopt five benchmark datasets for the evaluation, including (1) **MSRVTT** [55] that contains 10K video clips, where each has 20 captions. We follow the 1K-A testing split [34]. (2) **LSMDC** [45] incorporating 118081 clips from 202 movies, where each one is paired with a text description. Following [15, 17], we adopt the testing data with 1000 videos. (3) **DiDeMo** [2] consists of 10642 clips and 40543 captions in total. We use the training/testing data following [26, 39]. (4) **Charades** [46] contains 9848 video clips, where each corresponds to a text description. We adopt the same split protocol as in [33]. (5) **VATEX** [50] consists of 34, 991 video clips, where each corresponds to multiple text descriptions. We follow the train-test split of [6]. Recall at rank $\{1, 5, 10\}$ (R@1, R@5, and R@10), Median Rank (MdR), and Mean Rank (MnR) are adopted to evaluate the retrieval performance.

Implementation Details. We employ X-Pool [17] as baseline. Both backbone models of CLIP [44] (both ViT-B/32 and ViT-B/16) are leveraged for the feature extraction, following previous methods [17, 57]. We keep the configurations the same as X-Pool, such that setting dimension d = 512, weight decay as 0.2, and dropout as 0.3. For the training, we set the batch size as 32 for both backbones

[1] See supplementary material for more discussions about stochastic text embedding and KL-divergence .

Method	MSRVTT Retrieval					LSMDC Retrieval				
	R@1 ↑	R@5 ↑	R@10 ↑	MdR ↓	MnR ↓	R@1 ↑	R@5 ↑	R@10 ↑	MdR ↓	MnR ↓
CLIP-ViT-B/32										
X-Pool [17]	46.9	72.8	82.2	2.0	14.3	25.2	43.7	53.5	8.0	53.2
DiffusionRet [26]	49.0	75.2	82.7	2.0	12.1	24.4	43.1	54.3	8.0	**40.7**
UATVR [13]	47.5	73.9	83.5	2.0	12.3	–	–	–	–	–
TEFAL [21]	49.4	**75.9**	83.9	2.0	12.0	26.8	46.1	56.5	7.0	44.4
CLIP-ViP [57]	50.1	74.8	84.6	1.0	–	25.6	45.3	54.4	8.0	–
T-MASS (Ours)	**50.2**	75.3	**85.1**	**1.0**	**11.9**	**28.9**	**48.2**	**57.6**	**6.0**	43.3
CLIP-ViT-B/16										
X-Pool [17]	48.2	73.7	82.6	2.0	12.7	26.1	46.8	56.7	7.0	47.3
UATVR [13]	50.8	76.3	85.5	1.0	12.4	–	–	–	–	–
CLIP-ViP [57]	**54.2**	**77.2**	84.8	1.0	–	29.4	50.6	59.0	5.0	–
T-MASS (Ours)	52.7	77.1	**85.6**	**1.0**	**10.5**	**30.3**	**52.2**	**61.3**	**5.0**	**40.1**

Table 1. Text-to-video comparisons on MSRVTT [55] and LSMDC [45]. Bold denotes the best performance. "–": result is unavailable.

Method	DiDeMo Retrieval					VATEX Retrieval				
	R@1 ↑	R@5 ↑	R@10 ↑	MdR ↓	MnR ↓	R@1 ↑	R@5 ↑	R@10 ↑	MdR ↓	MnR ↓
CLIP-ViT-B/32										
X-Pool [17]	44.6	73.2	82.0	2.0	15.4	60.0	90.0	95.0	1.0	3.8
DiffusionRet [26]	46.7	74.7	82.7	2.0	14.3	–	–	–	–	–
UATVR [13]	43.1	71.8	82.3	2.0	15.1	61.3	91.0	95.6	1.0	3.3
CLIP-ViP [57]	48.6	77.1	84.4	2.0	–	–	–	–	–	–
T-MASS (Ours)	**50.9**	**77.2**	**85.3**	**1.0**	**12.1**	**63.0**	**92.3**	**96.4**	**1.0**	**3.2**
CLIP-ViT-B/16										
X-Pool [17]	47.3	74.8	82.8	2.0	14.2	62.6	91.7	96.0	1.0	3.4
UATVR [13]	45.8	73.7	83.3	2.0	13.5	64.5	92.6	96.8	1.0	2.8
CLIP-ViP [57]	50.5	78.4	87.1	1.0	–	–	–	–	–	–
T-MASS (Ours)	**53.3**	**80.1**	**87.7**	**1.0**	**9.8**	**65.6**	**93.9**	**97.2**	**1.0**	**2.7**

Table 2. Text-to-video comparisons on DiDeMo [2] and VATEX [50]. Bold denotes the best performance. "–": result is unavailable.

Method	R@1	R@5	R@10	MdR	MnR
CLIP-ViT-B/32					
CLIP4Clip [39]	42.7	70.9	80.6	2.0	11.6
CenterCLIP [60]	42.8	71.7	82.2	2.0	10.9
X-Pool [17]	44.4	73.3	84.0	2.0	9.0
TS2-Net [36]	45.3	74.1	83.7	2.0	9.2
DiffusionRet [26]	47.7	73.8	84.5	2.0	8.8
UATVR [13]	46.9	73.8	83.8	2.0	8.6
T-MASS (Ours)	**47.7**	**78.0**	**86.3**	**2.0**	**8.0**
CLIP-ViT-B/16					
X-Pool [17]	46.4	73.9	84.1	2.0	8.4
TS2-Net [36]	46.6	75.9	84.9	2.0	8.9
CenterCLIP [60]	47.7	75.0	83.3	2.0	10.2
UATVR [13]	48.1	76.3	85.4	2.0	8.0
T-MASS (Ours)	**50.9**	**80.2**	**88.0**	**1.0**	**7.4**

Table 3. Video-to-text comparisons on MSRVTT.

Method	R@1	R@5	R@10	MdR	MnR
CLIP-ViT-B/32					
ClipBERT [28]	6.7	17.3	25.2	32.0	149.7
CLIP4Clip [39]	9.9	27.1	36.8	21.0	85.4
X-Pool [17]	11.2	28.3	38.8	20.0	82.7
T-MASS (Ours)	**14.2**	**36.2**	**48.3**	**12.0**	**54.8**
CLIP-ViT-B/16					
CLIP4Clip [39]	16.0	38.2	48.5	12.0	54.1
X-Pool [17]	20.7	42.5	53.5	9.0	47.4
T-MASS (Ours)	**26.7**	**51.7**	**63.9**	**5.0**	**30.0**

Table 4. Text-to-video comparisons on Charades [46].

and different datasets. We keep the same initial learning rate of $1e-5$ to train the feature fusion module $\psi(\cdot)$ and the proposed similarity-aware radius module $\mathcal{R}$ ($3e-5$ for MSRVTT). The CLIP model is fine-tuned with a learning rate of $1e-6$. We train the models for 5 epochs with the AdamW [38] optimizer. Following CLIP, we employ a cosine schedule [37] with a warm-up proportion of 0.1. We uniformly sample 12 frames from the video clips upon different datasets. All the frames are resized to 224×224. We perform experiments on an A6000 GPU. We set sampling trials $T' = 20$ during inference. Some methods use larger batch size and larger frames numbers for different datasets. We keep it consistent for our method by using batch size as 32 and frame number as 12 for all datasets. More results and discussions are provided in supplementary.

4.2. Performance Comparison

We compare the text-to-video retrieval performance of T-MASS with previous methods on five benchmark datasets.

Radius $\mathcal{R}$	MSRVTT Retrieval					DiDeMo Retrieval				
	R@1 ↑	R@5 ↑	R@10 ↑	MdR ↓	MnR ↓	R@1 ↑	R@5 ↑	R@10 ↑	MdR ↓	MnR ↓
w/o $\mathcal{R}$	46.9	72.8	82.2	2.0	14.3	44.6	73.2	82.0	2.0	15.4
$\exp(\frac{1}{T'}\sum S_i)$	48.7	74.7	83.7	2.0	12.7	48.0	75.4	85.0	2.0	13.0
$\exp(\frac{\theta}{T'}\sum S_i)$	**49.2**	75.7	84.7	2.0	**11.7**	49.7	75.8	85.3	2.0	12.6
$\exp(\mathbf{SW})$	49.1	**75.7**	**85.7**	**2.0**	11.9	**49.8**	**78.1**	**86.0**	**2.0**	**11.8**

Table 5. Model discussion on similarity-aware radius module design. We perform experiments on MSRVTT and DiDeMo. CLIP-ViT-B/32 is adopted. Notebaly, "w/o $\mathcal{R}$" denotes the baseline of X-Pool [17]. We choose $\exp(\mathbf{SW})$ for the final performance comparison.

$\mathbf{t}_s$	$\mathcal{L}_{\text{ce}}$	$\mathcal{L}_s$	$\mathcal{L}_{\text{sup}}$	R@1	R@5	R@10	MdR	MnR
✗	✓	✗	✗	46.9	72.8	82.2	2.0	14.3
✓	✓	✓	✗	48.5	74.8	84.3	2.0	12.3
✓	✗	✓	✗	49.1	**75.7**	**85.7**	2.0	11.9
✓	✗	✓	✓	**50.2**	75.3	85.1	**1.0**	**11.9**

(a) Ablation study of losses and text embedding on MSRVTT [55].

#Trials (M)	R@1	R@5	R@10	MdR	MnR
w/o sampling	44.4	72.4	81.9	2.0	13.1
5	46.8	74.7	84.0	2.0	12.5
10	50.0	75.2	84.1	2.0	12.3
20	**50.2**	**75.3**	**85.1**	**1.0**	**11.9**

(b) Discussion of stochastic sampling trails on MSRVTT-1K.

Table 6. Discussion of the text representation, learning objectives, and number of trials. $\mathbf{t}_s$ denotes stochastic embedding, relative to text embedding $\mathbf{t}$. $\mathcal{L}_{\text{ce}}$ denotes symmetric cross entropy loss upon Eq. (2). $\mathcal{L}_s$ computes upon $\mathbf{t}_s$ and $\mathbf{v}$. $\mathcal{L}_{\text{sup}}$ denotes support text regularization.

We find that T-MASS not only improves the baseline X-Pool by a large margin on all metrics, but also achieves state-of-the-art performance compared with most recent methods. As shown in Table 1, T-MASS improves improves CLIP-ViP 3.3% at R@1 on LSMDC ViT-B/32 model. In Table 2, T-MASS improves X-Pool by 6.0% at R@1 on DiDeMo upon ViT-b/16. By observation, the proposed method shows a consistent performance boost on versatile datasets and different scales of model size. There exists one scenario under MSRVTT and ViT-B/16 that CLIP-ViT works better than T-MASS. Note that besides the retrieval data, CLIP-ViP also adopts additional datasets, *e.g.*, WebVid-2.5M [3] and HD-VILA-100M [56] to further empower the post-pretraining, potentially better adapt the CLIP. Employing more data especially benefits the larger model of ViT-B/16. T-MASS outperforms CLIP-ViP on other datasets and backbones. To save the computational cost, this work does not include the additional multi-modal data. As shown Table 3, T-MASS also enables the best performance for video-to-text retrieval. CLIP-ViP is skipped as the result is unavailable. In summary, since T-MASS empowers text embedding with more flexibility, it potentially explores more possibilities in text-video alignment. We provide a more in-depth analysis in the following.

4.3. Model Discussion

Similarity-Aware Radius. In Table 5, we provide three options to implement the similarity-aware radius module, as introduced in Section 3.3. Specifically, $\exp(\frac{1}{T'}\sum S_i)$ denotes only using cosine similarity to implement the radius, which is unlearnable. We further incorporate a learnable scalar θ, resulting $\exp(\frac{\theta}{T'}\sum S_i)$, or using a linear layer, yielding $\exp(\mathbf{SW})$. Note that "w/o $\mathcal{R}$" denotes the baseline of X-Pool [17]. The design of the $\mathcal{R}$ brings a clear performance boost compared with the baseline, *i.e.*, $> 1.5\%+$ at R@1 on MSRVTT and $> 3\%+$ at R@1 on DiDeMo. This indicates that representing text as a semantics range can indeed further benefit the retrieval on these two datasets, compared with $\mathbf{t}$. Besides, using a learnable module to can further boost the performance as the expressiveness and flexibility of the mass improve. The design of $\exp(\mathbf{SW})$ works best in most cases (especially on DiDeMo), owning to a stronger modeling capacity. Interestingly, only using a learnable scalar also enables strong performance, indicating that our approach is not sensitive to the network design of $\mathcal{R}$. We adopt $\exp(\mathbf{SW})$ in our final model.

Ablation Study. We provide an ablation study on MSRVTT in terms of text representation and learning objectives in Table 6a. Firstly, we show the baseline of X-Pool (top row). Based on X-Pool, we substitute text embedding $\mathbf{t}$ with $\mathbf{t}_s$ and correspondingly add a $\mathcal{L}_s$, obtaining 1.6% boost at R@1. This shows the superiority of $\mathbf{t}_s$ over $\mathbf{t}$. We further evaluate the effect of the original loss $\mathcal{L}_{\text{ce}}$ under the regime of the stochastic embedding $\mathbf{t}_s$. By comparison, highlighting the text embedding $\mathbf{t}$ with $\mathcal{L}_{\text{ce}}$ undermines the performance (2nd and 3rd rows of Table 6a). This is because further regularizing $\mathbf{t}$ can lead to a biased text mass learning, misleading the retrieval. Rather, we adopt a support text vector $\mathbf{t}_{\text{sup}}$ as a proxy to control the scale and shift of the text mass. Since $\mathbf{t}_{\text{sup}}$ locates at the surface of the text mass upon Eq. (8), as shown in Fig. 4, controlling support text embedding can affect the whole text mass. We adopt the last setting (the 4th row in Table 6a) in our final model.

Inference Discussion. We discuss the number of sampling trials for inference in Table 6b. For "w/o sampling", we still use the original $\mathbf{t}$ during the metric computation. This gives a sub-optimal performance as there is no exploitation of the text mass. As the number of trails M increases from 5 to 20, the proposed method enables better performance by exploiting more possibilities of the stochastic embedding $\mathbf{t}_s$ corresponding to the semantics of the raw text t. The number of 5 may not be enough to explore the

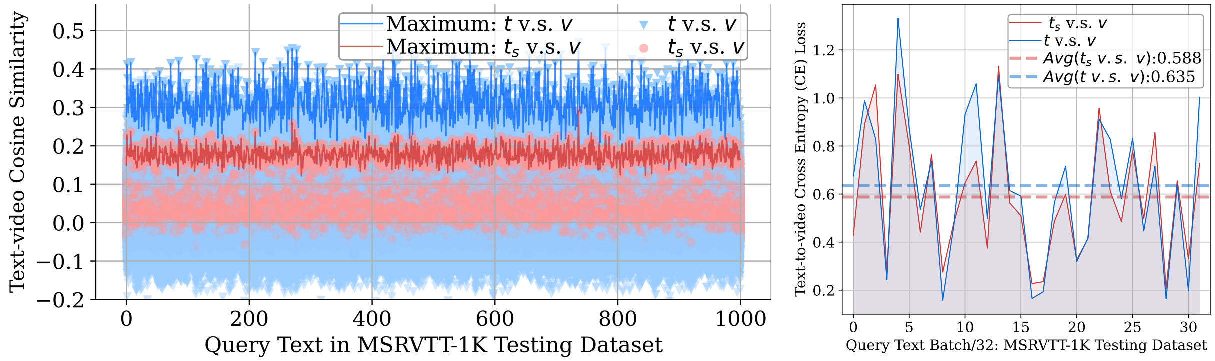

Figure 5. Analysis of stochastic text embedding $\mathbf{t}_s$, text embedding $\mathbf{t}$, and video embedding $\mathbf{v}$ in a joint space. ***Left***: Cosine similarities of **irrelevant** text-video pairs in embedding space. ***Right***: Cross entropy values of **relevant** text-video pairs in embedding space. The proposed stochastic text embedding allows a lower similarity for irrelevant pairs and enables lower cross entropy loss for relevant pairs.

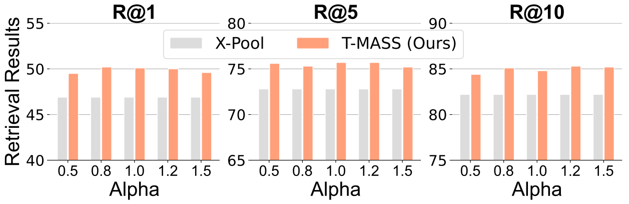

Figure 6. Discussion of support text regularization weight α. We compare T-MASS with baseline X-Pool [17] on MSRVTT [55].

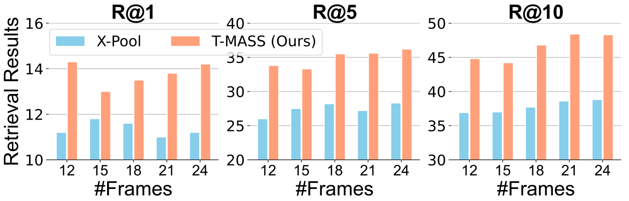

Figure 7. Performance boost under different #frames (T'). We compare T-MASS with the baseline X-Pool [17] on Charades [46].

text mass, leading to a sub-optimal result. Note that the performance tends to be stable from $M = 10$ to $M = 20$. We choose 20 in our final model, considering the trade-off between the performance and computational cost.

Further Analysis on T-MASS. We further analyze the behavior of the T-MASS by observing the stochastic text embedding $\mathbf{t}_s$, text embedding $\mathbf{t}$, and video embedding $\mathbf{v}$ in the same joint space. As shown in Fig. 5 *left*, we collect the text-video cosine similarity values for all irrelevant pairs in MSRVTT-1K, comparing between $\{\mathbf{t}^i \text{ v.s. } \mathbf{v}^j\}$ and $\{\mathbf{t}_s^i \text{ v.s. } \mathbf{v}^j\}$, where $i \neq j$. For each query text, we plot similarity values to all irrelevant videos (*i.e.*, 999) and highlight the maximum value (red and blue curves). The smaller similarity values are, the better irrelevant text-video pairs are aligned and thus potentially benefits the retrieval. By observation, using stochastic embedding $\mathbf{t}_s$ gives a better result than $\mathbf{t}$ (red curve is lower). This indicates that the irrelevant $\mathbf{t}$ and $\mathbf{v}$ can be close to each other, which may impose more risks for mismatching. We also visualize the cross-entropy loss values of relevant text-video pairs in Fig. 5 *right*. By average, the proposed $\mathbf{t}_s$ enables lower entropy, which reflects higher similarities for relevant t-v pairs, ensuring a more promising and accurate retrieval. In summary, both comparisons (Fig. 5 *left* and *right*) show that T-MASS indeed enables a better text-video embedding alignment.

Hyperparameter Discussion. Fig. 6 discusses the effect of the support text regularization. Retrieval performance under different penalties, such as $\alpha = \{0.5, 0.8, 1.0, 1.2, 1.5\}$ are presented. The performance of X-Pool is provided as a reference. We achieve the best performance at $\alpha = 1.2$. We also discuss the effect of the #frames on Charades in Fig. 7. Specifically, we report performance with $T' = \{12, 15, 18, 21, 24\}$. T-MASS enables a notable performance boost under different T' values. Owning to the text mass learning, this method demonstrates robustness to different configurations of input videos.

5. Conclusions

This work studied the text-video retrieval task by drawing attention to the text side – text is hard to fully describe the semantics of a video, implying that text embedding may not be expressive enough to capture or align to the video – based on which we opted to enrich the text embedding with more flexibility and resilience. We introduced T-MASS, where text is modeled as a stochastic embedding, facilitating joint learning of the text mass and video points. Our method incorporated similarity-aware radius modeling and a support text vector as regularization to better align relevant/irrelevant text-video pairs and encourage text semantics determination. Experiments on five datasets demonstrated that T-MASS achieved state-of-the-art performance. We hope this work will inspire future endeavors from the perspective of text in advancing text-video retrieval.

6. Acknowledgement

This research was partially supported by the DEVCOM Army Research Laboratory under Contract W911QX-21-D-0001.

References

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

[2] Lisa Anne Hendricks, Oliver Wang, Eli Shechtman, Josef Sivic, Trevor Darrell, and Bryan Russell. Localizing moments in video with natural language. In *ICCV*, 2017. 5, 6

[3] Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman. Frozen in time: A joint video and image encoder for end-to-end retrieval. In *ICCV*, 2021. 1, 2, 7

[4] Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman. A clip-hitchhiker's guide to long video retrieval. *arXiv*, 2022. 1

[5] Simion-Vlad Bogolin, Ioana Croitoru, Hailin Jin, Yang Liu, and Samuel Albanie. Cross modal retrieval with querybank normalisation. In *CVPR*, 2022. 1, 2

[6] Shizhe Chen, Yida Zhao, Qin Jin, and Qi Wu. Fine-grained video-text retrieval with hierarchical graph reasoning. In *CVPR*, 2020. 2, 5

[7] Xing Cheng, Hezheng Lin, Xiangyu Wu, Fan Yang, and Dong Shen. Improving video-text retrieval by multi-stream corpus alignment and dual softmax loss. *arXiv*, 2021. 1, 2

[8] Ioana Croitoru, Simion-Vlad Bogolin, Marius Leordeanu, Hailin Jin, Andrew Zisserman, Samuel Albanie, and Yang Liu. Teachtext: Crossmodal generalized distillation for text-video retrieval. In *CVPR*, 2021. 2

[9] Chaorui Deng, Qi Chen, Pengda Qin, Da Chen, and Qi Wu. Prompt switch: Efficient clip adaptation for text-video retrieval. In *ICCV*, 2023. 2

[10] Jianfeng Dong, Xianke Chen, Minsong Zhang, Xun Yang, Shujie Chen, Xirong Li, and Xun Wang. Partially relevant video retrieval. In *ACM MM*, 2022. 2

[11] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv*, 2020. 2

[12] Maksim Dzabraev, Maksim Kalashnikov, Stepan Komkov, and Aleksandr Petiushko. Mdmmt: Multidomain multimodal transformer for video retrieval. In *CVPR*, 2021. 2

[13] Bo Fang, Chang Liu, Yu Zhou, Min Yang, Yuxin Song, Fu Li, Weiping Wang, Xiangyang Ji, Wanli Ouyang, et al. Uatvr: Uncertainty-adaptive text-video retrieval. In *ICCV*, 2023. 1, 2, 6

[14] Han Fang, Pengfei Xiong, Luhui Xu, and Yu Chen. Clip2video: Mastering video-text retrieval via image clip. *arXiv preprint arXiv:2106.11097*, 2021. 2

[15] Valentin Gabeur, Chen Sun, Karteek Alahari, and Cordelia Schmid. Multi-modal transformer for video retrieval. In *ECCV*, 2020. 2, 3, 5

[16] Zijian Gao, Jingyu Liu, Sheng Chen, Dedan Chang, Hao Zhang, and Jinwei Yuan. CLIP2TV: an empirical study on transformer-based methods for video-text retrieval. *CoRR*, 2021. 2

[17] Satya Krishna Gorti, Noël Vouitsis, Junwei Ma, Keyvan Golestan, Maksims Volkovs, Animesh Garg, and Guangwei Yu. X-pool: Cross-modal language-video attention for text-video retrieval. In *CVPR*, 2022. 1, 2, 3, 5, 6, 7, 8

[18] Peiyan Guan, Renjing Pei, Bin Shao, Jianzhuang Liu, Weimian Li, Jiaxi Gu, Hang Xu, Songcen Xu, Youliang Yan, and Edmund Y Lam. Pidro: Parallel isomeric attention with dynamic routing for text-video retrieval. In *ICCV*, 2023. 1, 2

[19] Ning Han, Jingjing Chen, Hao Zhang, Huanwen Wang, and Hao Chen. Adversarial multi-grained embedding network for cross-modal text-video retrieval. *ACM MM*, 2022. 2

[20] J. Huang, Y. Li, J. Feng, X. Wu, X. Sun, and R. Ji. Clover: Towards a unified video-language alignment and fusion model. In *CVPR*, 2023. 2

[21] Sarah Ibrahimi, Xiaohang Sun, Pichao Wang, Amanmeet Garg, Ashutosh Sanan, and Mohamed Omar. Audio-enhanced text-to-video retrieval using text-conditioned feature alignment. In *ICCV*, 2023. 2, 6

[22] Y. Ji, R. Tu, J. Jiang, W. Kong, C. Cai, W. Zhao, H. Wang, Y. Yang, and W. Liu. Seeing what you miss: Vision-language pre-training with semantic completion learning. In *CVPR*, 2023. 2

[23] Peng Jin, Jinfa Huang, Fenglin Liu, Xian Wu, Shen Ge, Guoli Song, David Clifton, and Jie Chen. Expectation-maximization contrastive learning for compact video-and-language representations. In *NeurIPS*, 2022. 2

[24] Peng Jin, Jinfa Huang, Pengfei Xiong, Shangxuan Tian, Chang Liu, Xiangyang Ji, Li Yuan, and Jie Chen. Video-text as game players: Hierarchical banzhaf interaction for cross-modal representation learning. In *CVPR*, 2023. 1

[25] Peng Jin, Hao Li, Zesen Cheng, Jinfa Huang, Zhennan Wang, Li Yuan, Chang Liu, and Jie Chen. Text-video retrieval with disentangled conceptualization and set-to-set alignment. In *IJCAI*, 2023. 1

[26] Peng Jin, Hao Li, Zesen Cheng, Kehan Li, Xiangyang Ji, Chang Liu, Li Yuan, and Jie Chen. Diffusionret: Generative text-video retrieval with diffusion model. In *ICCV*, 2023. 2, 5, 6

[27] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013. 2, 3, 4

[28] Jie Lei, Linjie Li, Luowei Zhou, Zhe Gan, Tamara L Berg, Mohit Bansal, and Jingjing Liu. Less is more: Clipbert for video-and-language learning via sparse sampling. In *CVPR*, 2021. 2, 6

[29] Linjie Li, Zhe Gan, Kevin Lin, Chung-Ching Lin, Zicheng Liu, Ce Liu, and Lijuan Wang. Lavender: Unifying video-language understanding as masked language modeling. In *CVPR*, 2023. 2

[30] Pandeng Li, Chen-Wei Xie, Liming Zhao, Hongtao Xie, Jiannan Ge, Yun Zheng, Deli Zhao, and Yongdong Zhang. Progressive spatio-temporal prototype matching for text-video retrieval. In *ICCV*, 2023. 2

[31] Y. Li, K. Min, S. Tripathi, and N. Vasconcelos. Svitt: Temporal learning of sparse video-text transformers. In *CVPR*, 2023. 2

[32] Chengzhi Lin, Ancong Wu, Junwei Liang, Jun Zhang, Wenhang Ge, Wei-Shi Zheng, and Chunhua Shen. Text-adaptive multiple visual prototype matching for video-text retrieval. *NeurIPS*, 2022. 1

[33] Yan-Bo Lin, Jie Lei, Mohit Bansal, and Gedas Bertasius. Eclipse: Efficient long-range video retrieval using sight and sound. In *ECCV*, 2022. 2, 5

[34] Yang Liu, Samuel Albanie, Arsha Nagrani, and Andrew Zisserman. Use what you have: Video retrieval using representations from collaborative experts. In *BMVC*, 2019. 2, 5

[35] Yu Liu, Huai Chen, Lianghua Huang, Di Chen, Bin Wang, Pan Pan, and Lisheng Wang. Animating images to transfer clip for video-text retrieval. In *ACM SIGIR*, 2022. 2

[36] Yuqi Liu, Pengfei Xiong, Luhui Xu, Shengming Cao, and Qin Jin. Ts2-net: Token shift and selection transformer for text-video retrieval. In *ECCV*, 2022. 1, 2, 6

[37] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016. 6

[38] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 6

[39] Huaishao Luo, Lei Ji, Ming Zhong, Yang Chen, Wen Lei, Nan Duan, and Tianrui Li. Clip4clip: An empirical study of clip for end to end video clip retrieval and captioning. In *Neurocomputing*, 2022. 1, 2, 5, 6

[40] Antoine Miech, Ivan Laptev, and Josef Sivic. Learning a text-video embedding from incomplete and heterogeneous data. *arXiv*, 2018. 2

[41] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018. 3

[42] R. Pei, J. Liu, W. Li, B. Shao, S. Xu, P. Dai, J. Lu, and Y. Yan. Clipping: Distilling clip-based models with a student base for video-language retrieval. In *CVPR*, 2023. 2

[43] Jesús Andrés Portillo-Quintero, José Carlos Ortiz-Bayliss, and Hugo Terashima-Marín. A straightforward framework for video retrieval using clip. In *MCPR*, 2021. 1

[44] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, 2021. 1, 2, 3, 5

[45] Anna Rohrbach, Marcus Rohrbach, Niket Tandon, and Bernt Schiele. A dataset for movie description. In *CVPR*, 2015. 5, 6

[46] Gunnar A Sigurdsson, Gül Varol, Xiaolong Wang, Ali Farhadi, Ivan Laptev, and Abhinav Gupta. Hollywood in homes: Crowdsourcing data collection for activity understanding. In *ECCV*, 2016. 5, 6, 8

[47] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *NeurIPS*, 2017. 2

[48] Qiang Wang, Yanhao Zhang, Yun Zheng, Pan Pan, and Xian-Sheng Hua. Disentangled representation learning for text-video retrieval. *arXiv*, 2022. 1

[49] Wenzhe Wang, Mengdan Zhang, Runnan Chen, Guanyu Cai, Penghao Zhou, Pai Peng, Xiaowei Guo, Jian Wu, and Xing Sun. Dig into multi-modal cues for video retrieval with hierarchical alignment. In *IJCAI*, 2021. 1

[50] Xin Wang, Jiawei Wu, Junkun Chen, Lei Li, Yuan-Fang Wang, and William Yang Wang. Vatex: A large-scale, high-quality multilingual dataset for video-and-language research. In *ICCV*, 2019. 5, 6

[51] Xiaohan Wang, Linchao Zhu, and Yi Yang. T2vlad: global-local sequence alignment for text-video retrieval. In *CVPR*, 2021. 1, 2

[52] Ziyue Wang, Aozhu Chen, Fan Hu, and Xirong Li. Learn to understand negation in video retrieval. In *ACM MM*, 2022. 2

[53] Wenhao Wu, Haipeng Luo, Bo Fang, Jingdong Wang, and Wanli Ouyang. Cap4video: What can auxiliary captions do for text-video retrieval? In *CVPR*, 2023. 2

[54] Hu Xu, Gargi Ghosh, Po-Yao Huang, Dmytro Okhonko, Armen Aghajanyan, Florian Metze, Luke Zettlemoyer, and Christoph Feichtenhofer. VideoCLIP: Contrastive pre-training for zero-shot video-text understanding. In *EMNLP*, 2021. 2

[55] Jun Xu, Tao Mei, Ting Yao, and Yong Rui. Msr-vtt: A large video description dataset for bridging video and language. In *CVPR*, 2016. 5, 6, 7, 8

[56] Hongwei Xue, Tiankai Hang, Yanhong Zeng, Yuchong Sun, Bei Liu, Huan Yang, Jianlong Fu, and Baining Guo. Advancing high-resolution video-language representation with large-scale video transcriptions. In *CVPR*, 2022. 7

[57] Hongwei Xue, Yuchong Sun, Bei Liu, Jianlong Fu, Ruihua Song, Houqiang Li, and Jiebo Luo. Clip-vip: Adapting pre-trained image-text model to video-language representation alignment. In *ICLR*, 2023. 1, 2, 3, 5, 6

[58] Youngjae Yu, Jongseok Kim, and Gunhee Kim. A joint sequence fusion model for video question answering and retrieval. In *ECCV*, 2018. 2, 3

[59] N. Zhao, J. Jiao, W. Xie, and D. Lin. Cali-nce: Boosting cross-modal video representation learning with calibrated alignment. In *CVPRW*, 2023. 2

[60] Shuai Zhao, Linchao Zhu, Xiaohan Wang, and Yi Yang. Centerclip: Token clustering for efficient text-video retrieval. In *ACM SIGIR*, 2022. 1, 2, 6

[61] Linchao Zhu and Yi Yang. Actbert: Learning global-local video-text representations. In *CVPR*, 2020. 1