T2VLAD: Global-Local Sequence Alignment for Text-Video Retrieval

Xiaohan Wang[1,2*] Linchao Zhu[3] Yi Yang[3]
[1]Zhejiang University [2]Baidu Research [3]ReLER, University of Technology Sydney
wxh1996111@gmail.com Linchao.Zhu@uts.edu.au Yi.Yang@uts.edu.au

Abstract

Text-video retrieval is a challenging task that aims to search relevant video contents based on natural language descriptions. The key to this problem is to measure text-video similarities in a joint embedding space. However, most existing methods only consider the global cross-modal similarity and overlook the local details. Some works incorporate the local comparisons through cross-modal local matching and reasoning. These complex operations introduce tremendous computation. In this paper, we design an efficient global-local alignment method. The multi-modal video sequences and text features are adaptively aggregated with a set of shared semantic centers. The local cross-modal similarities are computed between the video feature and text feature within the same center. This design enables the meticulous local comparison and reduces the computational cost of the interaction between each text-video pair. Moreover, a global alignment method is proposed to provide a global cross-modal measurement that is complementary to the local perspective. The global aggregated visual features also provide additional supervision, which is indispensable to the optimization of the learnable semantic centers. We achieve consistent improvements on three standard text-video retrieval benchmarks and outperform the state-of-the-art by a clear margin.

1. Introduction

Video is one of the most informative media due to the abundant multi-modal content and temporal dynamics. Text-video retrieval systems enable humans to search videos with a simple and natural interaction approach. Recently, some efforts have been made in building retrieval systems with complex text inputs [2, 9], *e.g.*, retrieving contents of "a group of men inspect and test a brand new yellow car". This is more applicable as the users could search content based on more detailed descriptions.

One of the promising directions to enable cross-modal video retrieval is to measure text-video similarities using metric learning [34, 7]. In this case, the common practice is to embed both descriptions and videos into a joint embedding space. Most existing works [24] [6] [22] [9] encode the descriptions and video content to global representations and compare their similarities from a global perspective. These methods focus on the learning of effective language and video representations but overlook the fine-grained semantic alignment. For instance, Gabeur *et al.* [9] leveraged a multi-modal transformer to enhance the valuable cross-modal interaction to generate more discriminative video features. Some other works [2, 21, 32] leveraged complex cross-modal matching operations to exploit the local details and align multiple semantic cues. Chen *et al.* [2] proposed a hierarchical graph reasoning model to capture both global events and local actions through local graph matching. They manually designed three levels of semantics, including events, actions, and entities. However, these methods require a high computational cost due to the expensive pairwise matching operation.

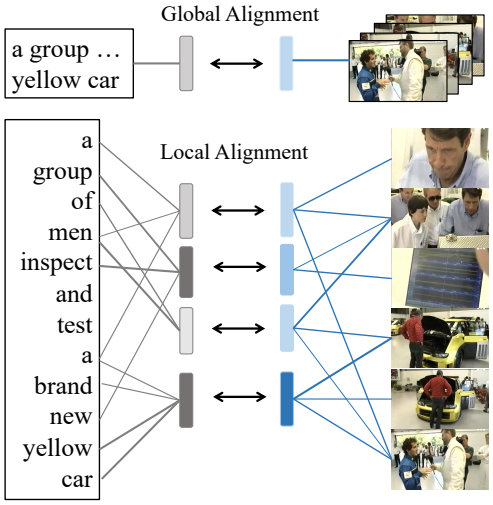

Figure 1. Global alignment gives a comprehensive similarity measurement between texts and videos. Local alignment provides fine-grained comparisons by computing the similarities between the local text-video features from the same semantic centers.

*Work done during an internship at Baidu Research.

In this paper, we propose an efficient global-local sequence alignment method for text-video retrieval. In the **local perspective**, we aim to utilize a number of learnable semantic topics to jointly summarize both texts and videos. Instead of parsing text descriptions to a hierarchical semantic role graph [2], it is hoped that these semantic topics could be discovered and automatically learned during the end-to-end training. We further share the weights of text topics and video topics to offer a joint topic representation learning and to reduce the semantic gap between text and video data. To achieve local alignment, we minimize the distance between the grouped text feature and the corresponding grouped video features within the same topics. In the **global perspective**, the multi-modal video sequences are aggregated temporally within each modality. The global similarity is computed between the aggregated video features and global text features. The global alignment not only serves as a complementary measurement to local alignment but also provides additional supervision for the learnable semantic topics.

We implement the idea of local semantic topic alignment with the help of a NetVLAD operation [1]. In NetVLAD, the learnable centers are regarded as "visual words" of the input data, which can be readily utilized as latent semantic topics on our cross-modal video retrieval task. For both text and video modalities, we use NetVLAD operations to obtain an aggregated feature for each topic, where the topic centers are **shared** between the two modalities. The text features and video features are softly assigned to topics based on their corresponded similarities. Without complex graph operations [2] and multi-layer transformers [9], we surprisingly find that our collaborative encoding method, namely Text-to-Video VLAD (T2VLAD), could boost the retrieval performance on various datasets. The contribution of this paper can be summarized as below:

- First, we propose to automatically learn text-and-video semantic topics and re-emphasize the importance of local semantic alignment between texts and videos for better cross-modal retrieval.

- Second, we introduce an effective strategy to locally align text inputs and video inputs. Based on the success of NetVLAD encoding [1], we propose a T2VLAD encoding for cross-modal retrieval, where we exploit shared centers to reduce the semantic gap between texts and videos instead of the complex pairwise local matching operation.

- Third, we demonstrate significant improvements of T2VLAD on three standard text-video retrieval benchmarks, *i.e.*, MSRVTT [35], ActivityNet Captions [19], and LSMDC [28]. Notably, we outperform a HowTo100M-pretrained [25] multi-modal transformer [9] with 2.9% gain (Rank@1) on MSRVTT without any additional data.

2. Related Work

Text-Video Retrieval. There are increasing interests in advancing text-video retrieval performance [27, 8, 2, 9]. Compared to text-image retrieval [7, 17, 16], text-video retrieval is more challenging that requires the understanding of temporal dynamics and complicated text semantics. A few works [27, 26] focus on visual semantic embedding learning for text and video joint modeling. Mithun *et al.* [26] leveraged a simple text-image embedding method [7] to improve the training strategy with hard negative mining, and incorporated multi-modal features (RBG, motion, and audio) to enrich the video representations. Dong *et al.* [6] proposed dual-encoding network with multiple levels of features for text-video retrieval, *i.e.*, features obtained by mean pooling, bi-directional Gated Recurrent Unit and Convolution Layers. Yu *et al.* [37] proposed a joint fusion model using Long Short-Term Memory for temporal sequential information encoding between videos and texts. Liu *et al.* [22] further utilize all modalities that can be extracted from videos such as speech contents and scene texts for video encoding. Miech *et al.* [24] introduced a strong joint embedding using mixture-of-expert features, which are later utilized in [9].

Language Representation Learning. Language representations are usually learned using sequence encoders, *e.g.*, Long Short-Term Memory [12], Gated Recurrent Unit [3]. Recently, with the success of BERT model [4] in contextual text representation learning using multi-layer transformer architectures [30], many vision-and-language works [9, 29, 43] leveraged pre-trained BERT features to enhance the language representation capability. Similar to [9], we use the BERT model during text-video retrieval and the model is fine-tuned during our end-to-end cross-modal retrieval training.

VLAD Encoding. VLAD [15] and NetVLAD [1] have achieved great impacts in aggregating discriminative features for video classification [10, 36], video retrieval [24], person re-identification [40]. NetVLAD is an end-to-end differentiable layer that could be readily plugged into many existing models. These works usually leverage the NetVLAD layer as a discriminative feature learner for downstream tasks. However, in this paper, we leverage NetVLAD in text-video local similarity matching and introduce a local alignment loss to reduce the gap of locally learned features from texts and videos. We do not conduct classification upon the obtained aggregated features, but apply local alignment between the text and video features.

3. Method

3.1. Overview

We propose Text-to-Video VLAD (T2VLAD) for cross-modal retrieval, which aligns text and video features in a global and local perspective. Given a text-video pair, our goal is to encode it into a joint feature space to measure the similarity. As shown in Fig. 2, we leverage multiple experts to extract the local video features corresponding to each modality (Section 3.2). The BERT model is utilized to extract contextual word features (Section 3.3). After that, we feed all the video features from different experts to a self-attention layer to enhance the features based on cross-modal relations. The output video features and text features are assigned to a set of cluster centers, which are shared between text encoding and video encoding. We aggregate the local features based on the assignments and generate the locally aligned features for both video and text to compute a local video-text similarity (Section 3.4). To provide additional supervision on the local alignment and introduce complementary information, we develop a global alignment scheme (Section 3.5).

3.2. Video Representations

Compared to image data, videos are more complex and contain richer information such as motion, audio and speech. To make full use of the multi-modal information in video data for the text-video retrieval task, we leverage multiple experts [24, 22, 9] to encode raw videos. Specifically, given an input video, we leverage N experts $\{\boldsymbol{E}^1, \boldsymbol{E}^2, \ldots, \boldsymbol{E}^N\}$ to extract multi-modal features. Here $\boldsymbol{E}^n$ represents the n-th expert. Each expert is pretrained on a particular task to acquire specific knowledge on the corresponding modal. Our goal is to achieve both local and global alignment for text-video retrieval, so we extract features from each temporal segment. For each expert, we obtain a set of segment-level video representations, *i.e.*, $\{\boldsymbol{E}^n(\boldsymbol{x}_1), \boldsymbol{E}^n(\boldsymbol{x}_2), \ldots, \boldsymbol{E}^n(\boldsymbol{x}_T)\}$. Here T is the number of segments, and $\boldsymbol{x}_t$ is the t-th segment from a video. We leverage the following two operations to further process the segment-level multi-expert features for the subsequent global-local alignment.

First, we introduce to generate **global expert features for global alignment**. We aim to perform temporal aggregation for each expert to generate global expert features. There are a few existing temporal aggregation operations to obtain a global vector, *e.g.*, temporal convolution networks [20], Transformers [30] and NetVLAD [1]. For simplicity, we leverage a max-pooling operation without additional parameters. This simple operation works well in our experiments. The temporal-aggregated features are projected to the same dimension for the subsequent clustering. Following [24], we then enhanced the features by a self-gating mechanism. Consequently, we obtain a set of global expert features $\{\boldsymbol{F}_1^{video}, \boldsymbol{F}_2^{video}, \ldots, \boldsymbol{F}_N^{video}\}$, where N is the number experts.

Second, we use one self-attention layer to **fuse multi-expert features for local alignment**. We first employ a fully-connected layer for each expert to project different expert features to a C-dimensional embedding space. We then concatenate the features from all experts to generate the local features $\boldsymbol{Z}^{video} = \{\boldsymbol{z}_1^{video}, \boldsymbol{z}_2^{video}, \ldots, \boldsymbol{z}_M^{video}\}$, where M is the number of features from all experts. We further explore the relations among the multi-modal features with self-attention mechanism. This design is similar to [9] but has two differences: (1) We only use an one-layer transformer encoder [30] instead of the multi-layer transformer with pre-aggregation and position encoding as in [9]. Thus, our module introduces fewer parameters and is more computationally efficient; (2) We aim to maintain the locality of the input features while [9] generates aggregated expert features for the subsequent text-to-video matching. The output feature $\boldsymbol{Z}^{video}$ of this process has the same length as the input features.

3.3. Text Representations

The BERT model [4] has shown great generalization capabilities in language feature encoding. We leverage a pre-trained BERT model to fairly compared to [9]. The BERT model extracts the contextual word embeddings for each text input. The input sentences are tokenized and padded to be a fixed-length sequence. The fixed-length sequence is the input to the BERT model. We add special tokens like "[CLS]" and "[SEP]" to indicate the start and the end of the sentence. The features can be computed as $\boldsymbol{Z}^{text} = \Phi^{BERT}(S)$, where Φ^{BERT} is the BERT model, S is the input tokens. $\boldsymbol{Z}^{text} = \{\boldsymbol{z}_1^{text}, \boldsymbol{z}_2^{text}, \ldots, \boldsymbol{z}_B^{text}\}$, where B is the sequence length. The BERT model Φ^{BERT} is optimized with the other modules in our framework in an end-to-end manner. It provides powerful text modeling capacity. Different from video encoding, the global features for text are extracted jointly with local representations for the subsequent T2VLAD module.

3.4. Local Alignment

After the aforementioned text encoding and video encoding, we obtain B local contextual word embeddings $\boldsymbol{Z}^{text}$ and M video local features $\boldsymbol{Z}^{video}$ for each input text-video pair. These features contain abundant information about the input sentences and videos. However, the direct comparisons between the two types of features are not feasible because they are not well-aligned. Moreover, the local video features $\boldsymbol{Z}^{video}$ are from different modalities. The domain gaps increase the difficulties of the local alignment. Intuitively, if we select and aggregate the local text features and video features on the same topic and

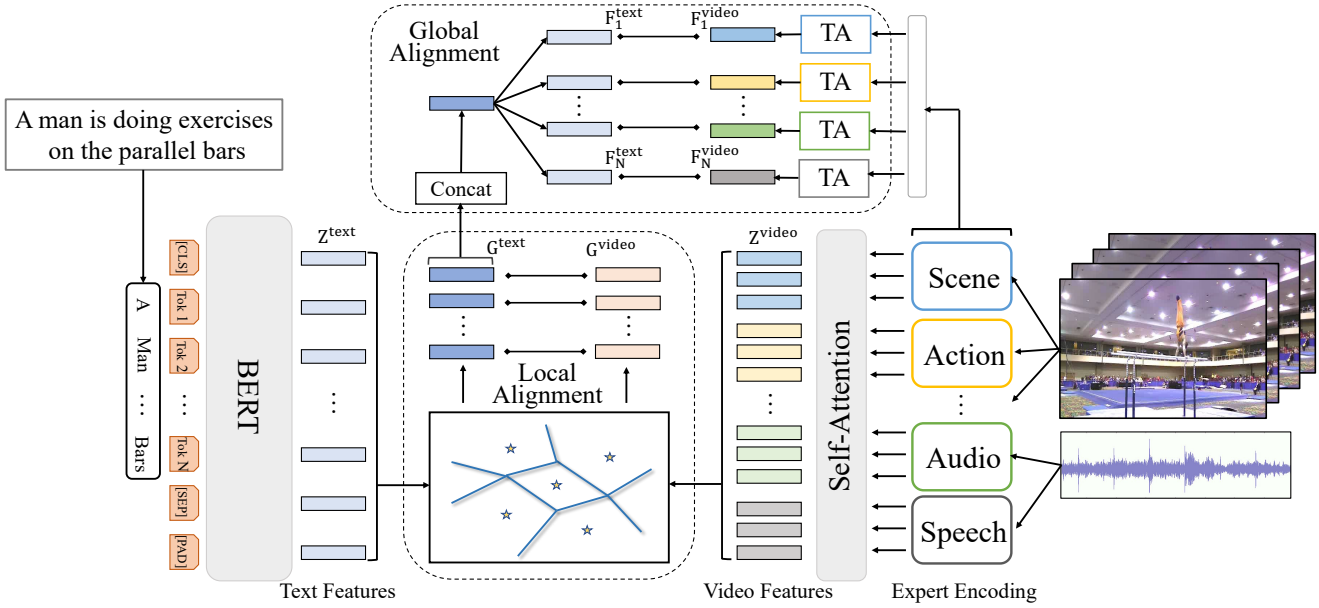

Figure 2. Our T2VLAD framework. "TA" indicates temporal aggregation. Given a text-video pair, we leverage multiple experts to extract the local video features corresponding to each modality. A BERT model is utilized to extract contextual word features. We feed all the video features from different experts to a self-attention layer to enhance the features based on cross-modal relations. The output video features and text features are assigned to a set of shared centers. We aggregate the local features based on the assignments and generate the locally aligned features for both video and text to compute a local video-text similarity. We develop a global alignment scheme in which the video features from each expert are aggregated to a global feature to calculate a similarity with the projected global text feature.

then compare their similarities, the measurement would become more precise. Motivated by this spirit, we propose Text-to-Video VLAD (T2VLAD) to cluster the local features from multiple modalities with shared centers. These centers provide shared semantic topics which can bridge the gaps among different modalities. Inspired by [1], these centers can be learned jointly with the whole network, and the feature clustering can be performed on-the-fly.

Specifically, we learn $K+1$ C-dimensional shared cluster centers $\{\boldsymbol{c}_1, \boldsymbol{c}_2, \ldots, \boldsymbol{c}_K, \boldsymbol{c}_{K+1}\}$. Here the K centers are for local alignment and the additional center is for background information removal. The design of the background center shares the same spirit of [41] to discard noise information. We follow [1] to calculate the similarities between each local feature and the cluster centers using dot-product. This step computes assignments on the corresponding clusters. We start with the encoding of video features. Given a local video feature $\boldsymbol{z}_i^{video}$, its assignments to j-th cluster can be generated as follows,

$$a_{i,j} = \frac{exp(\boldsymbol{z}_i^{video}\boldsymbol{c}_j^{\mathsf{T}} + b_j)}{\sum_{k=1}^{K+1} exp(\boldsymbol{z}_i^{video}\boldsymbol{c}_k^{\mathsf{T}} + b_k)}, \tag{1}$$

where b_j is a learnable bias term. In practice, one can replace the bias term with a batch normalization layer [14] which normalizes and shifts the activation by two built-in learnable parameters. Then the aggregated residual feature on each centers can be obtained,

$$\boldsymbol{g}_j^{video} = \text{normalize}(\sum_{i=1}^{M} a_{i,j}(\boldsymbol{z}_i^{video} - \boldsymbol{c}_j')), \tag{2}$$

where the $\boldsymbol{c}_j'$ is trainable weights that have the same size as $\boldsymbol{c}_j$, and "normalize" indicates a ℓ_2-normalization operation. The design of introducing two centers for each cluster has been proposed in [1] to increase the adaptation capability of the NetVLAD layer. We obtain a set of aggregated video feature $\boldsymbol{G}^{video} = \{\boldsymbol{g}_1^{video}, \boldsymbol{g}_2^{video}, \ldots, \boldsymbol{g}_K^{video}\}$. Each feature in $\boldsymbol{G}^{video}$ is the aligned local feature for the video. Note that the aggregated feature on the background center is abandoned and not involved in the following similarity measurement.

The aggregated text features can be calculated in the same way using the shared cluster centers.

$$\boldsymbol{g}_j^{text} = \text{normalize}(\sum_{i=1}^{B} \frac{exp(\boldsymbol{z}_i^{text}\boldsymbol{c}_j^{\mathsf{T}} + b_j)}{\sum_{k=1}^{K+1} exp(\boldsymbol{z}_i^{text}\boldsymbol{c}_k^{\mathsf{T}} + b_k)}(\boldsymbol{z}_i^{text} - \boldsymbol{c}_j')), \tag{3}$$

where $\boldsymbol{z}_i^{text}$ is the local word embedding in $\boldsymbol{Z}^{text}$. We can obtain the final local feature $\boldsymbol{G}^{text} = \{\boldsymbol{g}_1^{text}, \boldsymbol{g}_2^{text}, \ldots, \boldsymbol{g}_K^{text}\}$ for the text sequence. Since the local feature assignment and aggregation for video and text share the same centers, the final features $\boldsymbol{G}^{video}$ and $\boldsymbol{G}^{text}$

can be aligned effectively. We utilize cosine distance to measure the local similarity between the final video and text features $s_{local} = dist(\boldsymbol{G}^{video}, \boldsymbol{G}^{text})$.

3.5. Global Alignment

We introduce global alignment for two reasons. First, the global features for text-video pairs are more comprehensive and complementary to local features. Second, the elaborate local alignment with trainable centers can be difficult to be optimized when lacking auxiliary supervision, especially when the video features consist of multi-modal information.

Therefore, we alleviate the optimization difficulty in global alignment by aggregating and transforming the video feature from each expert independently. Meanwhile, we utilize the concatenation of local text features $\boldsymbol{G}^{text}$ to generate the expert-specific global text representations $\{\boldsymbol{F}_1^{text}, \boldsymbol{F}_1^{text}, \ldots, \boldsymbol{F}_N^{text}\}$. And each feature is then used to compute the similarity with the corresponding video expert feature. Following [24], we compute the global text-video similarity as a weighted sum of cosine distances between each global video expert feature and corresponding text feature. Formally, the global similarity is calculated as follows,

$$s_{global} = \sum_{i=1}^{N} w_i * dist(\boldsymbol{F}_i^{text}, \boldsymbol{F}_i^{video}), \tag{4}$$

where w_i is the weight for the i-th expert. The weights are generated from the text representation $\boldsymbol{G}^{text}$ by a linear projection with a softmax normalization. We utilize the text-video similarity $s = \frac{1}{2}(s_{global} + s_{local})$ to obtain a simple bi-directional max-margin ranking loss on both text-to-video and video-to-text retrieval tasks, following [24, 9]. We refer the reader to [24, 9] for detailed descriptions.

4. Experiments

4.1. Experimental Details

Dataset. We experiment with MSRVTT [35], video-text datasets. The **MSRVTT** dataset contains 10,000 videos. These videos are collected from YouTube using 257 queries from a commercial video search engine. We evaluate the performance on three splits. For the "1k-A" split, the train and test are split as introduced in [37]. The "1k-B" split is obtained following [24]. Both splits use 9,000 videos for training, and the remaining 1,000 videos are used for testing. The **ActivityNet Captions** dataset [19] consists of 20,000 videos. Each video is densely annotated with multiple sentence descriptions. The **LSMDC** dataset [28] consists of 118,081 short video clips. The videos are extracted from 202 long movies.

Evaluation Metrics. We report the results with the standard video retrieval metrics, *i.e.*, Rank K (R@K, higher is better), Median Rank (MdR, lower is better). We report R@1, R@5, and R@10 following [2, 22].

Multi-Expert Features. We use the features provided by [9] in our experiments. These features are: Motion features from S3D [33] trained on the Kinetics dataset. Audio features from VGGish model [11] trained on YT8M. Scene embeddings from DenseNet-161 [13] trained on the Places365 dataset [42]. We refer the readers to [9] for more descriptions of OCR, Face, Speech, and Appearance features. For MSRVTT, we also leverage optical flow features released by [9]. We do not use Speech features on LSMDC due to feature missing from the released features [9].

Implantation Details. We train the projection layer and the T2VLAD module from scratch, and no additional data is used. The margin in the ranking loss is set to 0.02 for all datasets. Following [22], we leverage Ranger optimizer with a weight decay 0.0001. We initialize the learning rate at 0.0001, and decay by a multiplicative factor 0.9 every 5 epochs. The batch size of the video-text pairs is set to 64. For text encoding, we use the pretrained BERT model "BERT-base-uncased" and fine-tune it with our framework in an end-to-end manner. For video expert encoding, we leverage the pre-extracted expert features provided by [9]. We use all 8 experts for the MSRVTT dataset and 6 experts (rgb, audio, ocr, scene, flow and action) for the LSMDC dataset. For ActivityNet Captions, we only use motion and audio experts. The self-attention module used for local video features is implemented by one layer multi-head attention with 4 heads, a dropout probability of 0.1, and a hidden size of 768. The dimension for the common space of both global alignment and local alignment is also set to 768. We set the center size of our T2VLAD to 9 for the short video retrieval dataset (MSRVTT and LSMDC) and 16 for the long video retrieval dataset (ActivityNet Captions).

4.2. Comparison to State-of-the-art

MSRVTT. The results on MSRVTT are shown in Table 1. We consistently improve the state-of-the-art on text-to-video retrieval and video-to-text retrieval across all three splits. MMT [9] is recently proposed to perform text-video retrieval using multi-modal transformers. It achieved the best performance in the compared methods. Notably, for text-to-video retrieval, we outperform MMT [9] with 5.8% gain on the R@1 metric on the 1k-B split (20.3% *vs*. 26.1%). A 5.6% improvement on R@1 (1k-B split) is also obtained compared to MMT [9] for video-to-text retrieval (21.1% *vs*. 26.7%). These results demonstrate the benefits of our T2VLAD in cross-modal retrieval tasks. Notably, we obtain consistent improvements over "MMT + HT pretrain" [9] on the 1k-A split. "MMT + HT pretrain" is pre-trained on a large-scale instructional video dataset, *i.e.*, HowTo100M, containing more than one hundred million video clips with machine-generated descriptions. Pre-

Method	Split	Text → Video				Video → Text			
		R@1↑	R@5↑	R@10↑	MdR↓	R@1↑	R@5↑	R@10↑	MdR↓
JSFusion [37]	1k-A	10.2	31.2	43.2	13	-	-	-	-
HT [25]	1k-A	14.9	40.2	52.8	9	-	-	-	-
CE [22]	1k-A	20.9	48.8	62.4	6	20.6	50.3	64.0	5.3
MMT [9]	1k-A	24.6	54.0	67.1	4	24.4	56.0	67.8	4
MMT + HT pretrain [9]	1k-A	26.6	57.1	69.6	4	27.0	57.5	69.7	3.7
Our T2VLAD	1k-A	**29.5**	**59.0**	**70.1**	**4**	**31.8**	**60.0**	**71.1**	**3**
MEE [24]	1k-B	13.6	37.9	51.0	10	-	-	-	-
JPose [31]	1k-B	14.3	38.1	53.0	9	16.4	41.3	54.4	8.7
MEE-COCO [24]	1k-B	14.2	39.2	53.8	9	-	-	-	-
CE [22]	1k-B	18.2	46.0	60.7	7	18.0	46.0	60.3	6.5
MMT [9]	1k-B	20.3	49.1	63.9	6	21.1	49.4	63.2	6
Our T2VLAD	1k-B	**26.1**	**54.7**	**68.1**	**4**	**26.7**	**56.1**	**70.4**	**4**
VSE [26]	Full	5.0	16.4	24.6	47	7.7	20.3	31.2	28
VSE++ [26]	Full	5.7	17.1	24.8	65	10.2	25.4	35.1	25
Mithun *et al.* [26]	Full	7.0	20.9	29.7	38	12.5	32.1	42.4	16
W2VV [5]	Full	6.1	18.7	27.5	45	11.8	28.9	39.1	21
Dual Enc. [6]	Full	7.7	22.0	31.8	32	13.0	30.8	43.3	15
HGR [2]	Full	9.2	26.2	36.5	24	15.0	36.7	48.8	11
E2E [23]	Full	9.9	24.0	32.4	29.5	-	-	-	-
CE [22]	Full	10.0	29.0	41.2	16	15.6	40.9	55.2	8.3
Our T2VLAD	Full	**12.7**	**34.8**	**47.1**	**12**	**20.7**	**48.9**	**62.1**	**6**

Table 1. The comparison with the state-of-the-art methods on the MSRVTT [35] dataset.

Method	Text → Video				Video → Text			
	R@1 ↑	R@5 ↑	R@50 ↑	MdR ↓	R@1 ↑	R@5 ↑	R@50 ↑	MdR ↓
FSE [39]	18.2	44.8	89.1	7	16.7	43.1	88.4	7
CE [22]	18.2	47.7	91.4	6	17.7	46.6	90.9	6
HSE [39]	20.5	49.3	-	-	18.7	48.1	-	-
MMT [9]	22.7	54.2	93.2	5	22.9	54.8	93.1	4.3
Ours	**23.7**	**55.5**	**93.5**	**4**	**24.1**	**56.6**	**94.1**	**4**

Table 2. The comparisons with the state-of-the-art methods on the ActivityNet Captions dataset.

training on HowTo100M significantly improves the performance of MMT across all evaluation metrics. T2VLAD does not leverage additional training videos, but we outperform "MMT + HT pretrain" on split 1k-A with a clear margin across all metrics. For instance, on text-to-video retrieval, T2VLAD outperforms "MMT + HT pretrain" by 2.9% at R@1. These results demonstrate that the benefit of the global-local alignment using T2VLAD.

The efficiency of our method is demonstrated by calculating inference time for 1k videos and 1k text queries from MSRVTT on a single V100 GPU. Our video encoding module (except expert encoding) only takes 0.4s for process 1k videos while MMT takes 1.1s. This shows the superiority of our efficient T2VLAD design.

ActivityNet Captions. ActivityNet Captions consists of long videos and the captions contain several sentences. The results on this dataset are shown in Table 2. The compared baselines include HSE [2], CE [22], HSE [39], and MMT [9]. HSE [39] leverages a hierarchical sequence embedding and MMT incorporates multi-layer transformers for strong video feature learning. We consistently improve MMT over all benchmark metrics, which demonstrates the effectiveness of T2VLAD on long-term text-video modeling.

LSMDC. The LSMDC data is collected from movies. The results are shown in Table 3. We observe consistent improvements over MMT. For instance, we achieve 2.1% improvements on R@1 for video-to-text retrieval. The results show that our T2VLAD is capable of dealing with different videos from different domains.

4.3. Ablation Study

The effectiveness of the global-local alignment. In Table 4, we show the results of only using the single alignment of our model. To implement the model without local alignment, we follow [9] to utilize the "[CLS]" output of the BERT model as the global text representation. When we remove the local alignment branch and only train the global alignment, the test performance drops a lot compared to the results of our full model. This proves our local alignment is crucial for the cross-modal retrieval task. When we remove the global alignment and only train the local alignment, the

Method	Text → Video				Video → Text			
	R@1 ↑	R@5 ↑	R@50 ↑	MdR ↓	R@1 ↑	R@5 ↑	R@50 ↑	MdR ↓
CT-SAN [38]	5.1	16.3	25.2	46	-	-	-	-
JSFusion [37]	9.1	21.2	34.1	36	-	-	-	-
CCA [18]	7.5	21.7	31.0	33	-	-	-	-
MEE [24]	9.3	25.1	33.4	27	-	-	-	-
MEE-COCO [24]	10.1	25.6	34.6	27	-	-	-	-
CE [22]	11.2	26.9	34.8	25.3	-	-	-	-
MMT [9]	13.2	29.2	38.8	21	12.1	29.3	37.9	22.5
Ours	**14.3**	**32.4**	**42.2**	**16**	**14.2**	**33.5**	**41.7**	**17**

Table 3. The comparison with the state-of-the-art methods on the LSMDC dataset.

Method	Text → Video				Video → Text			
	R@1↑	R@5↑	R@10↑	MdR↓	R@1↑	R@5↑	R@10↑	MdR↓
Ours w/o Global Alignment	24.3	51.5	63.4	5	26.6	52.9	62.6	5
Ours w/o Local Alignment	22.2	49.9	64.6	6	24.0	51.7	65.6	5
Full model	**29.5**	**59.0**	**70.1**	**4**	**31.8**	**60.0**	**71.1**	**3**

Table 4. The ablation studies on the MSRVTT [35] dataset to investigate the effectiveness of global-local alignment.

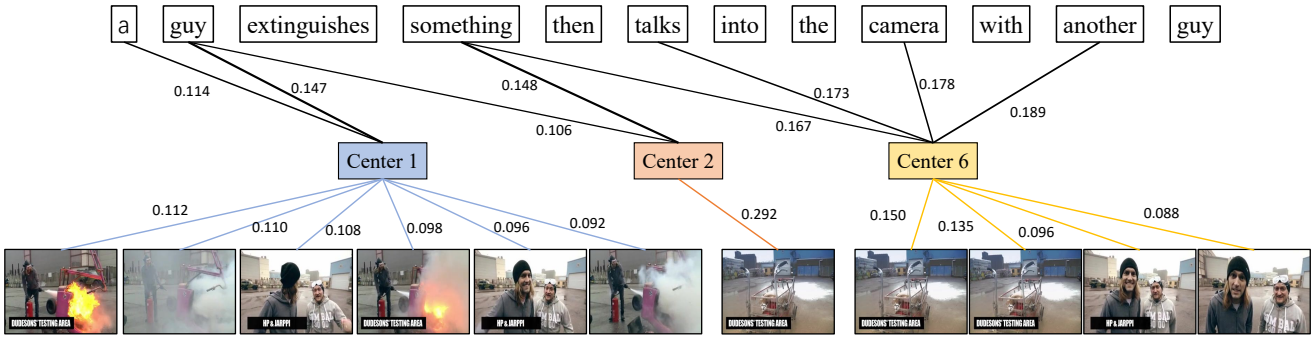

Figure 3. Visualization of the assignment weights. We take Video 7060 in the MSRVTT 1K-A test set as an example. We plot the top text assignments to the three centers as black lines and put the assignment values next to the line. The Top-10 frames (the padding features have been removed.) correspond to the appearance features assigned to the centers are shown at the bottom.

loss can not converge. It demonstrates the importance of global alignment for providing additional supervision for the optimization of the local alignment. We show the results of removing the global alignment only at test time, *i.e.*, "Ours w/o Global Alignment" in Table 4. Compared to the full model, the results drop by 5.2% on R@1 for text-to-video retrieval. It demonstrates that the global feature is complementary to the local information.

The effectiveness of collaborative VLAD. In "Ours w/ only text VLAD", we replace the shared NetVLAD layer for local video feature encoding with a max-pooling operation and then project the feature to the same dimension of text local features. This model achieves lower performance than our T2VLAD, showing the importance of joint VLAD encoding. In "Ours w/ two separate VLAD", we do not perform center sharing between text feature encoding and video feature encoding. The VLAD centers are learned separately. The results show that our strategy of sharing centers outperform "Ours w/ two separate VLAD" especially for text-to-video retrieval. This demonstrates that our center sharing idea is beneficial to reduce the semantic gap between text and video data.

4.4. Qualitative Results

Visualization of the assignments. The text local features and the local video features are assigned to a set of shared centers in our T2VLAD. We expect the aggregated text feature and video feature on the same center to share a similar topic. In Fig. 3, we illustrate the text assignments and video appearance feature assignments on three centers. The video is ranked first in the text retrieved results. We show the video frames corresponding to the appearance features that are assigned to the certain center. As shown in Fig. 3, the text feature with the highest assignment on Center 1 is the feature of "guy". All the frames that have been assigned to Center 1 also contain the appearance information of "guy". The text with the highest assignment on Center 2 is "something", and the only frame assigned to the center is about the "something" in the video. On Center 6, the text "something", "talks", "camera" and "another" all have high assignments. And the frames assigned to the center contain these content. Interestingly, the most salient word "extin-

Method	Text → Video				Video → Text			
	R@1↑	R@5↑	R@10↑	MdR↓	R@1↑	R@5↑	R@10↑	MdR↓
Ours w/ only text VLAD	27.4	57.3	68.2	4	27.5	57.4	69.7	4
Ours w/ two separate VLAD	28.6	58.1	70.4	4	30.4	60.7	72.1	3
Ours w/ two shared VLAD	**29.5**	**59.0**	70.1	**4**	**31.8**	60.0	71.1	**3**

Table 5. The ablation studies on the MSRVTT [35] dataset to investigate the effectiveness of the VLAD encoding.

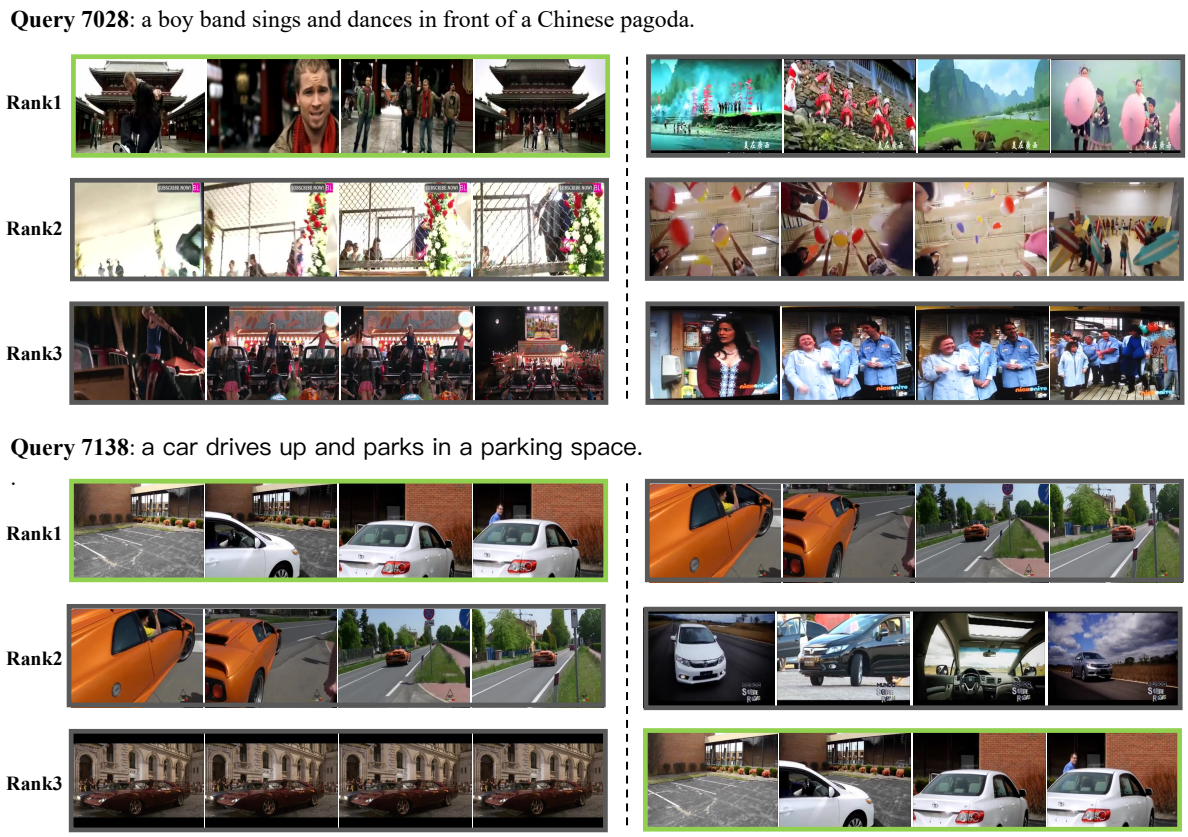

Figure 4. The text-video retrieval results on the MSRVTT 1K-A test set. The left are the videos ranked by our T2VLAD, and the right are the results from the model with only global alignment.

guishes" in the human view, always has a low assignment value on all centers. This is because the limited training data is not enough to enable the understanding of a low-frequency word. The assignment visualization verifies that our T2VLAD can achieve adequate local alignment for text-to-video retrieval.

Visualization of the text-to-video results. We show two examples of the videos retrieved by our method and the model without the local alignment branch. As shown in Fig. 4, the two query sentences consist of multiple semantic topics. Our T2VLAD successfully retrieves the ground-truth video while the model without local alignment returns several videos that are somewhat relevant to the query sentence but are not precise. In the second example, our T2VLAD achieves a better alignment between the text and videos on the local semantic cue "parks". These results demonstrate that our T2VLAD can align multiple semantic cues effectively.

5. Conclusion

In this paper, we introduce an end-to-end text-video sequence alignment method. We show that local semantic alignment between texts and videos is critical for high-performance retrieval systems. We achieve the goal of local alignment based on NetVLAD and introduce T2VLAD for collaborative text-video encoding. The results on three standard text-video retrieval benchmarks clearly demonstrate the effectiveness of our method. The visualization results also validate our motivation for joint semantic topic learning. In the future, more efforts could be paid to obtain better global video features with end-to-end optimization.

Acknowledgement Thanks to Samuel Albanie for his kind help with dataset processing and to Shizhe Chen for valuable discussions.

References

[1] Relja Arandjelovic, Petr Gronat, Akihiko Torii, Tomas Pajdla, and Josef Sivic. Netvlad: Cnn architecture for weakly supervised place recognition. In *CVPR*, 2016. 2, 3, 4

[2] Shizhe Chen, Yida Zhao, Qin Jin, and Qi Wu. Fine-grained video-text retrieval with hierarchical graph reasoning. In *CVPR*, 2020. 1, 2, 5, 6

[3] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*, 2014. 2

[4] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. 2, 3

[5] Jianfeng Dong, Xirong Li, and Cees GM Snoek. Predicting visual features from text for image and video caption retrieval. *IEEE Transactions on Multimedia*, 2018. 6

[6] Jianfeng Dong, Xirong Li, Chaoxi Xu, Shouling Ji, Yuan He, Gang Yang, and Xun Wang. Dual encoding for zero-example video retrieval. In *CVPR*, 2019. 1, 2, 6

[7] 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. 1, 2

[8] Hehe Fan and Yi Yang. Person tube retrieval via language description. In *AAAI*, 2020. 2

[9] Valentin Gabeur, Chen Sun, Karteek Alahari, and Cordelia Schmid. Multi-modal transformer for video retrieval. In *ECCV*, 2020. 1, 2, 3, 5, 6, 7

[10] Rohit Girdhar, Deva Ramanan, Abhinav Gupta, Josef Sivic, and Bryan Russell. Actionvlad: Learning spatio-temporal aggregation for action classification. In *CVPR*, 2017. 2

[11] Shawn Hershey, Sourish Chaudhuri, Daniel PW Ellis, Jort F Gemmeke, Aren Jansen, R Channing Moore, Manoj Plakal, Devin Platt, Rif A Saurous, Bryan Seybold, et al. Cnn architectures for large-scale audio classification. In *ICASSP*, 2017. 5

[12] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 1997. 2

[13] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *CVPR*, 2017. 5

[14] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015. 4

[15] Hervé Jégou, Matthijs Douze, Cordelia Schmid, and Patrick Pérez. Aggregating local descriptors into a compact image representation. In *CVPR*, 2010. 2

[16] Andrej Karpathy and Li Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In *CVPR*, 2015. 2

[17] Ryan Kiros, Ruslan Salakhutdinov, and Richard S Zemel. Unifying visual-semantic embeddings with multimodal neural language models. *arXiv preprint arXiv:1411.2539*, 2014. 2

[18] Benjamin Klein, Guy Lev, Gil Sadeh, and Lior Wolf. Associating neural word embeddings with deep image representations using fisher vectors. In *CVPR*, 2015. 7

[19] Ranjay Krishna, Kenji Hata, Frederic Ren, Li Fei-Fei, and Juan Carlos Niebles. Dense-captioning events in videos. In *ICCV*, 2017. 2, 5

[20] Colin Lea, Michael D Flynn, Rene Vidal, Austin Reiter, and Gregory D Hager. Temporal convolutional networks for action segmentation and detection. In *CVPR*, 2017. 3

[21] Kuang-Huei Lee, Xi Chen, Gang Hua, Houdong Hu, and Xiaodong He. Stacked cross attention for image-text matching. In *ECCV*, 2018. 1

[22] Y. Liu, S. Albanie, A. Nagrani, and A. Zisserman. Use what you have: Video retrieval using representations from collaborative experts. In *BMVC*, 2019. 1, 2, 3, 5, 6, 7

[23] Antoine Miech, Jean-Baptiste Alayrac, Lucas Smaira, Ivan Laptev, Josef Sivic, and Andrew Zisserman. End-to-end learning of visual representations from uncurated instructional videos. In *CVPR*, 2020. 6

[24] Antoine Miech, Ivan Laptev, and Josef Sivic. Learning a text-video embedding from incomplete and heterogeneous data. *arXiv preprint arXiv:1804.02516*, 2018. 1, 2, 3, 5, 6, 7

[25] Antoine Miech, Dimitri Zhukov, Jean-Baptiste Alayrac, Makarand Tapaswi, Ivan Laptev, and Josef Sivic. Howto100m: Learning a text-video embedding by watching hundred million narrated video clips. In *ICCV*, 2019. 2, 6

[26] 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 *ICMR*, 2018. 2, 6

[27] Yingwei Pan, Tao Mei, Ting Yao, Houqiang Li, and Yong Rui. Jointly modeling embedding and translation to bridge video and language. In *CVPR*, 2016. 2

[28] Anna Rohrbach, Marcus Rohrbach, Niket Tandon, and Bernt Schiele. A dataset for movie description. In *CVPR*, 2015. 2, 5

[29] Chen Sun, Austin Myers, Carl Vondrick, Kevin Murphy, and Cordelia Schmid. Videobert: A joint model for video and language representation learning. In *ICCV*, 2019. 2

[30] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017. 2, 3

[31] Michael Wray, Diane Larlus, Gabriela Csurka, and Dima Damen. Fine-grained action retrieval through multiple parts-of-speech embeddings. In *ICCV*, 2019. 6

[32] Yu Wu, Linchao Zhu, Yan Yan, and Yi Yang. Dual attention matching for audio-visual event localization. In *ICCV*, 2019. 1

[33] Saining Xie, Chen Sun, Jonathan Huang, Zhuowen Tu, and Kevin Murphy. Rethinking spatiotemporal feature learning: Speed-accuracy trade-offs in video classification. In *ECCV*, 2018. 5

[34] Eric P Xing, Michael I Jordan, Stuart J Russell, and Andrew Y Ng. Distance metric learning with application to clustering with side-information. In *NeurIPS*, 2003. 1

[35] Jun Xu, Tao Mei, Ting Yao, and Yong Rui. Msr-vtt: A large video description dataset for bridging video and language. In *CVPR*, 2016. 2, 5, 6, 7, 8

[36] Zhongwen Xu, Yi Yang, and Alex G Hauptmann. A discriminative cnn video representation for event detection. In *CVPR*, 2015. 2

[37] Youngjae Yu, Jongseok Kim, and Gunhee Kim. A joint sequence fusion model for video question answering and retrieval. In *ECCV*, 2018. 2, 5, 6, 7

[38] Youngjae Yu, Hyungjin Ko, Jongwook Choi, and Gunhee Kim. End-to-end concept word detection for video captioning, retrieval, and question answering. In *CVPR*, 2017. 7

[39] Bowen Zhang, Hexiang Hu, and Fei Sha. Cross-modal and hierarchical modeling of video and text. In *ECCV*, 2018. 6

[40] Liang Zheng, Yi Yang, and Qi Tian. Sift meets cnn: A decade survey of instance retrieval. *TPAMI*, 2017. 2

[41] Yujie Zhong, Relja Arandjelović, and Andrew Zisserman. Ghostvlad for set-based face recognition. In *ACCV*, 2018. 4

[42] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database for scene recognition. *TPAMI*, 2017. 5

[43] Linchao Zhu and Yi Yang. Actbert: Learning global-local video-text representations. In *CVPR*, 2020. 2