Find and Focus: Retrieve and Localize Video Events with Natural Language Queries

Dian Shao[1](✉), Yu Xiong[1], Yue Zhao[1], Qingqiu Huang[1], Yu Qiao[2], and Dahua Lin[1]

[1] CUHK-SenseTime Joint Lab, The Chinese University of Hong Kong, Shatin, Hong Kong
{sd017,xy017,zy317,hq016,dhlin}@ie.cuhk.edu.hk

[2] SIAT-SenseTime Joint Lab, Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Beijing, China
yu.qiao@siat.ac.cn

Abstract. The thriving of video sharing services brings new challenges to video retrieval, *e.g.* the rapid growth in video duration and content diversity. Meeting such challenges calls for new techniques that can effectively retrieve videos with natural language queries. Existing methods along this line, which mostly rely on embedding videos as a whole, remain far from satisfactory for real-world applications due to the limited expressive power. In this work, we aim to move beyond this limitation by delving into the internal structures of both sides, the queries and the videos. Specifically, we propose a new framework called *Find and Focus* (*FIFO*), which not only performs top-level matching (paragraph vs. video), but also makes part-level associations, localizing a video clip for each sentence in the query with the help of a focusing guide. These levels are complementary – the top-level matching narrows the search while the part-level localization refines the results. On both ActivityNet Captions and modified LSMDC datasets, the proposed framework achieves remarkable performance gains (Project Page: https://ycxioooong.github.io/projects/fifo).

1 Introduction

Over the past few years, the explosive growth of video content brings unprecedented challenges to video retrieval. Retrieving a video that one really wants is sometimes like finding a needle in a haystack. For example, entering a short query *"dancing people"* on Youtube would result in tens of millions of video

D. Shao and Y. Xiong—Equal contribution.

Electronic supplementary material The online version of this chapter (https://doi.org/10.1007/978-3-030-01240-3_13) contains supplementary material, which is available to authorized users.

V. Ferrari et al. (Eds.): ECCV 2018, LNCS 11213, pp. 202–218, 2018.
https://doi.org/10.1007/978-3-030-01240-3_13

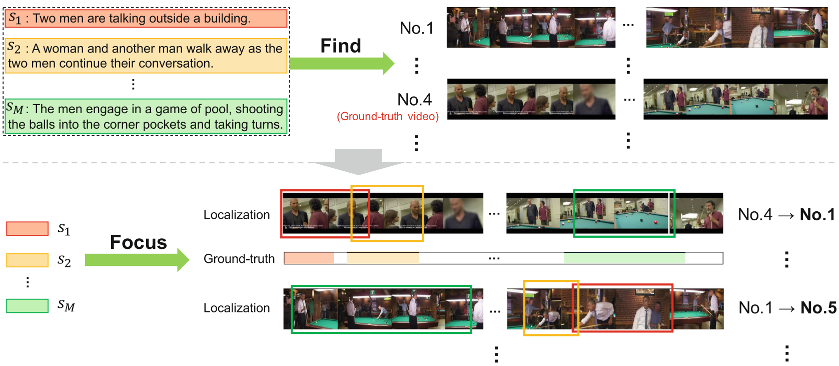

s_1 : Two men are talking outside a building.
s_2 : A woman and another man walk away as the two men continue their conversation.
s_M : The men engage in a game of pool, shooting the balls into the corner pockets and taking turns.
Find
No.1
No.4
(Ground-truth video)
s_1
s_2
s_M
Focus
Localization
Ground-truth
Localization
No.4 → No.1
No.1 → No.5

Fig. 1. An overview of our *Find and Focus* framework. Given a query paragraph, the system first retrieves a number of candidate videos in the Find stage, and then applies clip localization to each candidate video, to identify the associations between query sentences and video clips. The resulting localization scores can further refine the initial retrieval results. For example, the ground-truth video is ranked as No. 4 in the Find stage and promoted to No. 1 after the Focus stage.

entries, many of which are lengthy and filled with irrelevant fragments. To tackle such challenges, we aim to explore a new way to retrieve videos, one that can efficiently locate the relevant clips from a large and diverse collection.

Video retrieval is not new in computer vision. The research on this topic dates back to 1990s [26]. Classical content-based retrieval techniques [2,5,27,34,42] primarily rely on matching visual features with a fixed set of concepts. This approach can only work with a closed setting, where all videos belong to a predefined list of categories. The problem of *video retrieval in the wild* remains widely open. In recent years, an alternative approach, namely retrieving videos with natural language queries, emerges as a promising way to break the closed-set assumption. The efforts along this line are usually based on visual semantic embedding [6,7,13,16,20,30,36,38], where each image or video and its corresponding description are embedded into a common space and their representations are aligned.

It is noteworthy that both the classical techniques and visual semantic embedding share a common paradigm, namely, to encode each video as a whole into a feature vector and perform the retrieval simply by feature matching. This paradigm has two important limitations. First, a *single* vector representation lacks the expressive power to characterize a video with rich structures, and second, it lacks the capability of temporal localization, Note that these are not serious issues in conventional experimental settings where all video samples in the dataset are short clips. However, they become significant challenges in real-world applications where the videos are usually long and not trimmed.

In this work, we aim to move beyond such limits and develop an effective method that can retrieve complex events, *i.e.* those with rich temporal structures, based on natural language queries. We observe that people often describe a complex event with a paragraph, where each sentence may refer to a certain part of the event. This suggests that the association between a video and a relevant description exists not only at the top level but also between parts, *i.e.* sentences and video segments. With this intuition in mind, we explore a new idea, that is, to delve into the internal structures of both the queries and the videos, trying to identify and leverage the connections between their parts.

Specifically, we propose a structured framework to connect between the visual and the linguistic domains. The framework comprises two levels of associations, the *top-level* that matches the query paragraphs with whole videos, and the *part-level* that aligns individual sentences with video clips. On top of this formulation, we develop a two-stage framework called *Find and Focus* (*FIFO*), as shown in Fig. 1. Given a paragraph query, it first finds a subset of candidate videos via top-level matching. Then for each candidate, it localizes the clips for individual sentences in the query. Finally, the part-level associations are used to refine the ranking of retrieval results. In this way, the framework *jointly* accomplishes two tasks: retrieving videos and localizing relevant segments. Note that in our framework, these two tasks benefit each other. On one hand, the top-level matching narrows the search, thus reducing the overall cost, especially when working with a large database. On the other hand, the part-level localization refines the results, thus further improving the ranking accuracy. To facilitate clip localization, we develop a semantics-guided method to generate clip proposals, which allows the framework to focus on those clips with significant meanings.

Our main contributions are summarized as follows: (1) We propose a structured formulation that captures the associations between the visual and the linguistic domains at both top-level and part-level. (2) Leveraging the two-level associations, we develop a *Find and Focus* framework that jointly accomplishes video retrieval and clip localization. Particularly, the localization stage is supported by a new method, *Visual Semantic Similarity* (VSS), for proposing clip candidates, which helps to focus on the segments with significant meanings. (3) On two public datasets, ActivityNet Captions [17] and a modified version of Large Scale Movie Description Challenge (LSMDC) [23], the proposed framework obtains remarkable improvement.

2 Related Work

Visual Semantic Embedding. VSE [7,16] is a general approach to bridge visual and linguistic modalities. It has been adopted in various tasks, such as image question answering [22], image captioning [13,14], and image-text matching [6,16,31,36], etc. This approach was later extended to videos [19,21,24]. Plummer *et al.* [21] proposed to improve video summarization by learning a space for joint vision-language embedding. Zhu *et al.* [44] adopted the joint embedding method for aligning books to movies. In these works, each video is embedded as a whole, and its internal structures are not explicitly exploited.

Video Retrieval. Recent methods for video retrieval roughly fall into three categories: concept-based [2,5,27,34], graph-based [18], and those based on feature embeddings. Early works [27] often adopted the concept-based method, which involves detecting a list of visual concepts from the given videos. Recently, Yu *et al.* [41] proposed to improve this paradigm through end-to-end learning. A fundamental limitation of such methods is that they require a predefined list of concepts, which is difficult to provide sufficient coverage in real-world applications. Graph-based methods have also been widely used for matching images with text [11,12,37]. Lin *et al.* [18] explored a graph-based method which matches the objects in a video and the words in a description via bipartite matching. This method also requires a predefined list of objects and nouns.

Many works focused more on learning a joint embedding space for both videos and descriptions [20,30,38]. However, Otani *et al.* [20] embedded each video as a whole, therefore having difficulty in handling long videos that contain multiple events. It is not capable of temporal localization either. Also both [20] and [38] harness external resources through web search, while our framework only utilizes the video-text data in the training set. There are also works [3,4,29] aligning text and video based on character identities, discriminative clustering, or object discovery, without fully mining the semantic meaning of data.

Temporal Localization. Temporal localization, *i.e.* finding video segments for a query, is often explored in the context of action detection. Early methods mainly relied on sliding windows and hand-crafted features [8,10,28]. Recent works [25,40,43] improved the performance using convolutional networks. In these methods, *actionness* is a key factor to consider when evaluating proposals. However, in our settings, the query sentences can describe static scenes. Hence, we have to consider the *significance* of each proposal in a more general sense.

Retrieval in Video Captioning. We note that recent works on video captioning [17,39] often use video retrieval to assess the quality of generated captions. In their experiments, individual sentences and video clips are matched respectively. The temporal structures among video clips are not explicitly leveraged. Hence, these works essentially differ from our two-level structured framework.

3 Methodology

Our primary goal is to develop a framework that can retrieve videos with natural language descriptions and at the same time localize the relevant segments. For this task, it is crucial to model the temporal structures of the videos, for which only the top-level embeddings may not be sufficient. As mentioned, our basic idea is to delve into the internal structures, establishing the connections between the textual queries and the videos, not only at the top level, but also at the part level, *i.e.* sentences and video clips.

In this section, we formalize the intuition above into a two-level formulation in Sect. 3.1, which lays the conceptual foundation. We then proceed to describe how we identify the part-level associations between sentences and video clips in Sect. 3.2, which we refer to as *clip localization.* In Sect. 3.3, we put individual pieces together to form a new framework called *Find and Focus* (*FIFO*), which jointly accomplishes retrieval and localization.

3.1 Two-level Structured Formulation

Our task involves two domains: the query paragraphs in the linguistic domain and the videos in the visual domain. Both paragraphs and videos consist of internal structures. As shown in Fig. 2, a paragraph P is composed of a sequence of sentences $(s_1, \ldots, s_M)$; while a video V is composed of multiple clips $\{c_1, \ldots, c_N\}$, each capturing an event. When a paragraph P is describing a video V, each sentence s_i thereof may refer to a specific clip in V. We refer to such correspondences between sentences and clips as *part-level associations*. The part-level associations convey significant information about the relations between a video and a corresponding paragraph. As we will show in our experiments, leveraging such information can significantly improve the accuracy of retrieval.

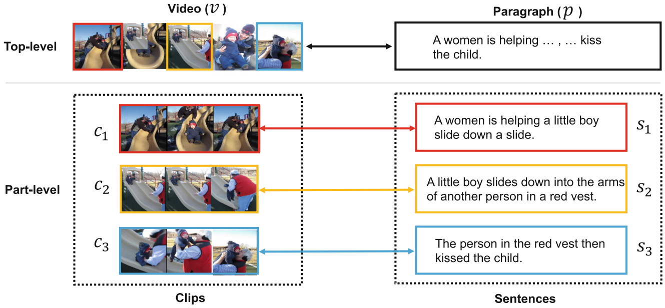

Video (v)
Paragraph (p)
Top-level
A women is helping … , … kiss the child.
Part-level
c_1
c_2
c_3
A women is helping a little boy slide down a slide.
A little boy slides down into the arms of another person in a red vest.
The person in the red vest then kissed the child.
s_1
s_2
s_3
Clips
Sentences

Fig. 2. This figure shows our two-level structured formulation. The upper half depicts the video-paragraph correspondence while the lower half represents the part-level associations between individual clips and sentences. Each individual pair of clip and sentence is denoted in different colors. (Color figure online)

3.2 Clip Localization

The part-level associations are identified via *clip localization*. Given a paragraph P and a video V, it first derives the features for the sentences in P and the snippets in V. Based on these features, it generates a collection of video clip candidates in a semantic-sensitive way, and then solves the correspondences between the sentences and the clips, via a robust matching method. The whole process of clip localization is illustrated in Fig. 3.

Feature Extraction. Given a video, it can be represented by a sequence of snippet-specific features as $V = (\mathbf{f}_1, \ldots, \mathbf{f}_T)$, where T is the number of snippets. The snippets are the units for video analysis. For every snippet (6 frames in our work), $\mathbf{f}_j$ is extracted with a two-stream CNN, trained following the TSN paradigm [35]. In a similar way, we can represent a query paragraph with a series of sentence-specific features as $P = (\mathbf{s}_1, \ldots, \mathbf{s}_M)$, where M is the number of sentences. Note that the visual features and the sentence features are in two *separate* spaces of different dimensions. To directly measure their similarities, we should first embed both features into a *common* semantic space as $\tilde{\mathbf{f}}_j$ and $\tilde{\mathbf{s}}_i$, where they are well aligned. The complete feature embedding process will be introduced in Sect. 3.3.

Clip Proposal. In our two-level formulation, each sentence corresponds to a video clip. A clip usually covers a range of snippets, and the duration of the clips for different sentences can vary significantly. Hence, to establish the part-level associations, we have to prepare a pool of clip candidates.

Inspired by the *Temporal Actionness Grouping* (TAG) method in [43], we develop a semantic-sensitive method for generating video clip proposals. The underlying idea is to find those continuous temporal regions, *i.e.* continuous ranges of snippets, that are semantically relevant to the queries. Specifically, given a sentence s_i, we can compute the *semantic relevance* of the j-th snippet by taking the cosine similarity between $\tilde{\mathbf{f}}_j$ and $\tilde{\mathbf{s}}_i$. Following the watershed scheme in TAG [43], we group the snippets into ranges of varying durations and thus obtain a collection of video clips[2]. For a query paragraph P, the entire clip pool is formed by the union of the collections derived for individual sentences.

Compared to TAG [43], the above method differs in how it evaluates the significance of a snippet. TAG is based on *actionness*, which is semantic-neutral and is only sensitive to those moments where certain actions happen; while our method uses semantic relevance, which is query-dependent and can respond to a much broader range of scenarios, including stationary scenes.

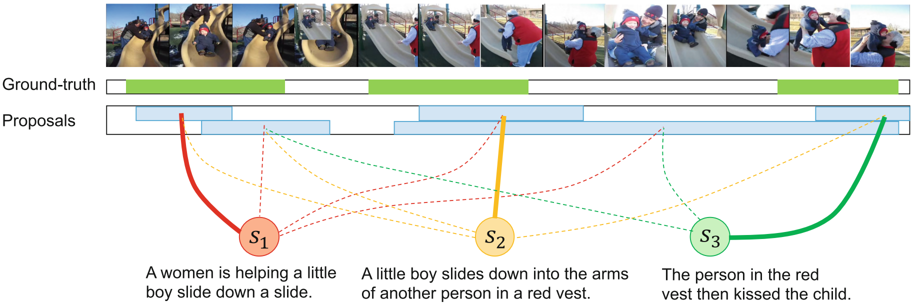

Ground-truth
Proposals
s_1
s_2
s_3
A women is helping a little boy slide down a slide.
A little boy slides down into the arms of another person in a red vest.
The person in the red vest then kissed the child.

Fig. 3. This figure shows the clip localization process. Given a video with ground-truth clips in green bars, a number of clip proposals colored in blue are generated using a semantic sensitive method. Each sentence is possibly associated with multiple clips, which are represented by thin dash lines. The optimal correspondence, illustrated by the thick lines, is obtained by a robust cross-domain matching. (Color figure online)

Cross-Domain Matching. Given a set of sentences $\{s_1, \ldots, s_M\}$ from the query paragraph P and a set of clip proposals $\{c_1, \ldots, c_N\}$ derived by the proposal generation method, the next is to find the correspondences between them.

[2] The technical details of this scheme is provided in the supplemental materials.

In principle, this can be accomplished by *bipartite matching*. However, we found empirically that the one-to-one correspondence enforced by bipartite matching can sometimes lead to misleading results due to outliers. To improve the robustness of the matching, we propose a *robust bipartite matching scheme*, which allows each sentence to be associated with up to u_{max} clips.

We can formalize this modified matching problem as a linear programming problem as follows. We use a binary variable x_{ij} to indicate the association between c_j and s_i. Then the problem can be expressed as

$$\text{maximize} \sum_{i=1}^{M}\sum_{j=1}^{N} r_{ij}x_{ij}; \qquad \text{s.t.} \ \sum_{j=1}^{N} x_{ij} \leq u_{max}, \ \forall i; \ \sum_{i=1}^{M} x_{ij} \leq 1, \ \forall j. \tag{1}$$

Here, r_{ij} is the *semantic relevance* between the sentence s_i the clip c_j, which is given by

$$r_{ij} \triangleq \frac{\tilde{\mathbf{s}}_i^T \tilde{\mathbf{g}}_j}{\|\tilde{\mathbf{s}}_i\| \cdot \|\tilde{\mathbf{g}}_j\|}, \quad \text{with} \ \tilde{\mathbf{g}}_j = \frac{1}{|C_j|}\sum_{t \in C_j} \tilde{\mathbf{f}}_t. \tag{2}$$

Here, $\tilde{\mathbf{g}}_j$ is the visual feature that summarizes the video clip c_j, which is snippet-wise feature averaged over its temporal window C_j. Moreover, the two inequalities in Eq. (1) respectively enforce the following constraints: (1) each sentence s_j can be matched to *at most* u_{max} clips, and (2) each clip corresponds to *at most* one sentence, *i.e.* the associated clips for different sentences are disjoint.

The above problem can be solved efficiently by Hungarian algorithm. The optimal value of the clip localization objective in Eq. (1) reflects how well the parts in both modalities can be matched. We call this optimal value *part-level association score*, and denote it by $S_p(V, P)$.

3.3 Overall Framework

Given a paragraph P, we can evaluate its relevance to each individual video by clip localization as presented above and thus obtain a ranked list of results, in descending order of the relevance score $S_p(V, P)$. However, this approach is prohibitively expensive, especially when retrieving from a large-scale database, as it requires performing proposal generation and solving the matching problem *on the fly*.

To balance the retrieval performance and runtime efficiency, we propose a two-stage framework called *Find and Focus*, which is illustrated in Fig. 1. In the Find stage, we perform top-level matching based on the overall representations for both the videos and the query. We found that while top-level matching may not be very accurate for ranking the videos, it can effectively narrow down the search by filtering out a majority of the videos in the database that are clearly irrelevant, while retaining most relevant ones. Note that top-level matching can be done very efficiently, as the top-level representations of the videos can be precomputed and stored. In the Focus stage, we perform detailed clip localization for each video in the top-K list by looking into their internal structures.

The resultant localization scores will be used to refine the ranking. The detailed procedure is presented below.

Find: Top-Level Retrieval. Given the snippet-level features denoted in Sect. 3.2, both the top-level representation $\mathbf{v}$ for a video V and $\mathbf{p}$ for a paragraph P can be achieved by aggregating all their part-level features.

In order to establish the connections between $\mathbf{v}$ and $\mathbf{p}$, at first we have to learn two embedding networks F_{vis}^{top} and F_{text}^{top} respectively for the visual and the linguistic domains, through which we could project them into a common space, as $\tilde{\mathbf{v}} = F_{vis}^{top}(\mathbf{v}; \mathbf{W}_{vis}^{top})$ and $\tilde{\mathbf{p}} = F_{text}^{top}(\mathbf{p}; \mathbf{W}_{text}^{top})$. Here, the embedding networks F_{vis}^{top} and F_{text}^{top} for top-level data can be learned based on the ranking loss [6,16]. Then the top-level relevance between V and P, denoted by $S_t(V, P)$, is defined as the cosine similarity between $\tilde{\mathbf{v}}$ and $\tilde{\mathbf{p}}$.

Based on the top-level relevance scores, we can pick the top K videos given a query paragraph P. We found that with a small K, the initial search can already achieve a high recall. Particularly, for ActivityNet Captions [17], which comprises about 5000 videos, the initial search can retain over 90% of the ground-truth videos in the top-K list with $K = 100$ (about 2% of the database).

Focus: Part-Level Refinement. Recall that through the embeddings learned in the Find stage, both visual features and linguistic features have already been projected into a common space Ω. These preliminarily embedded features could be further refined for clip localization task. The refined features for a snippet-specific feature f_j and a sentence s_i are denoted as $\tilde{\mathbf{f}}_j = F_{vis}^{ref}(F_{vis}^{top}(\mathbf{f}_j))$ and $\tilde{\mathbf{s}}_i = F_{text}^{ref}(F_{text}^{top}(\mathbf{s}_i))$, where F_{vis}^{ref} and F_{text}^{ref} represent the feature refinement networks. We will elaborate on how these feature embedding networks F^{top} and refinement networks F^{ref} are trained in Sect. 4.

For each of the K videos retained by the Find stage, we perform clip localization, in order to identify the associations between its clips and the sentences in the query. The localization process not only finds the clips that are relevant to a specific query sentence but also yields a part-level association score $S_p(V, P)$ for the video V at the same time.

Here, the part-level score $S_p(V, P)$, which is derived by aligning the internal structures, provides a more accurate assessment of how well the video V matches the query P and thus is a good complement to the top-level score $S_t(V, P)$. In this framework, we combine both scores into the *final* relevance score in a multiplicative way, as $S_r(V, P) = S_t(V, P) \cdot S_p(V, P)$. We use the final scores to re-rank the videos. Intuitively, this reflects the criterion that a truly relevant video should match the query at both the top level and the part level.

4 Learning the Embedding Networks

Our *Find and Focus* framework comprises two stages. In the first stage, a top-level embedding model is used to align the top-level features of both domains. In the second stage, the embedded features will be further refined for making part-level associations. Below we introduce how these models are trained.

Embedding for Top-Level Data. The objective of the first stage is to learn the networks F_{vis}^{top} and F_{text}^{top}, which respectively embed the original visual features $\{\mathbf{v}_j\}$ and the paragraph features $\{\mathbf{p}_i\}$ into a common space, as $\tilde{\mathbf{v}}_j = F_{vis}^{top}(\mathbf{v}_j; \mathbf{W}_{vis}^{top})$ and $\tilde{\mathbf{p}}_i = F_{text}^{top}(\mathbf{p}_i; \mathbf{W}_{text}^{top})$. These networks are learned jointly with the following margin-based ranking loss:

$$\mathcal{L}^{Find}(\mathbf{W}_{vis}^{top}, \mathbf{W}_{text}^{top}) = \sum_{i} \sum_{j \neq i} \max\left(0, S_t(V_j, P_i) - S_t(V_i, P_i) + \alpha\right). \quad (3)$$

Here, $S_t(V_j, P_i)$ is the top-level relevance between the video V_j and the paragraph P_i, which, as mentioned, is defined to be the cosine similarity between $\tilde{\mathbf{v}}_j$ and $\tilde{\mathbf{p}}_i$ in the learned space. Also, α is the margin which we set to 0.2. This objective encourages high relevance scores between each video and its corresponding paragraph, *i.e.* $S_t(V_i, P_i)$, and low relevance scores for mismatched pairs.

Refined Embedding for Part-Level Data. We use refined embeddings for identifying part-level associations. Specifically, given a clip c_j and a sentence s_i, their refined features, respectively denoted as $\tilde{\mathbf{g}}_j$ and $\tilde{\mathbf{s}}_i$, can be derived via refined embedding networks as follows:

$$\tilde{\mathbf{g}}_j = F_{vis}^{ref}(F_{vis}^{top}(\mathbf{g}_j; \mathbf{W}_{vis}^{top}); \mathbf{W}_{vis}^{ref}); \quad \tilde{\mathbf{s}}_i = F_{text}^{ref}(F_{text}^{top}(\mathbf{s}_i; \mathbf{W}_{text}^{top}); \mathbf{W}_{text}^{ref}). \quad (4)$$

Given s in a paragraph, we randomly pick one positive clip c^+ whose temporal IoU (tIoU) is greater than 0.7 out of all clip proposals from the corresponding video, and L negative proposals with tIoU below 0.3. The refined embedding networks F_{vis}^{ref} and F_{text}^{ref} are then trained with a ranking loss defined as below:

$$\mathcal{L}^{Ref}(\mathbf{W}_{vis}^{ref}, \mathbf{W}_{text}^{ref}) = \sum_{j=1}^{L} \max\left(0, s_r(c_j, s) - s_r(c^+, s) + \beta\right). \quad (5)$$

Here, $s_r(c_j, s)$ is the cosine similarity between the refined features as $s_r(c_j, s) = \cos(\tilde{\mathbf{g}}_j, \tilde{\mathbf{s}})$; and the margin β is set to 0.1. This loss function encourages high similarity between the embedded feature of the positive proposal c^+ and that of the query sentence s, while trying to reduce those between negative pairs.

5 Experiments

5.1 Dataset

ActivityNet Captions. ActivityNet Captions [17] consists of $20K$ videos with $100K$ sentences, which are aligned to localized clips. On average, each paragraph has 3.65 sentences, The number of annotated clips in one video ranges from 2 to 27, and the temporal extent of each video clip ranges from 0.05 s to 407 s. About 10% of the clips overlap with others. The complete dataset is divided into three disjoint subsets (training, validation, and test) by 2:1:1. We train models on the training set. Since the test set is not released, we test the learned models on the validation set `val_1`.

Modified LSMDC. LSMDC [23] consists of more than 128k clip-description pairs collected from 200 movies. However, for a considerable fraction of these movies, the provided clip descriptions are not well aligned with our acquired film videos possibly due to different versions. Excluding such videos and those kept for blind test, we retain 74 movies in our experiments. Besides, if we treat each movie as a video, we only have 74 video samples, which are not enough for training the top-level embedding. To circumvent this issue, we divide each movie into 3-min chunks, each serving as a whole video. In this way, 1677 videos are obtained and partitioned into two disjoint sets, 1188 videos from 49 movies for training and 489 videos from the other 25 movies for testing.

5.2 Implementation Details

For ActivityNet Captions, we extract a 1024-dimensional vector for every snippet of a video as its raw feature, using a TSN [35] with BN-Inception as its backbone architecture. We also extract word frequency histogram (Bag of Words weighted with tf-idf) as the raw representation for each paragraph or sentence. For the modified LSMDC, we use the feature from the pool5 layer of ResNet101 [9] as the raw feature for video data, and the sum of word embeddings for text.

We set the dimension of the common embedding space to be 512. We train both the top-level embedding networks in the Find stage and the refinement network in the Focus stage using Adam [15] with the momentum set to 0.9.

5.3 Whole Video Retrieval

We first compare our framework with the following methods on the task of whole video retrieval: (1) LSTM-YT [33] uses the latent states in the LSTM for cross-modality matching. (2) S2VT [32] uses several LSTMs to encode video frames and associate videos with text data. (3) Krishna *et al.* [17] encode each paragraph using the captioning model and each clip with a proposal model.

For performance evaluation, we employ the following metrics: (1) Recall@K, the percentage of ground truth videos that appear in the resultant top-K list, and (2) MedR, the median rank of the ground truth videos. These metrics are commonly used in retrieval tasks [17,20].

Table 1 shows the results of whole video retrieval on ActivityNet Captions dataset. From the results, we observe: (1) The VSE model trained in the Find stage is already able to achieve a substantial improvement over previous methods in terms of Recall@50, which shows that it is suitable for top-level matching. (2) Our proposed FIFO framework achieves the best performance consistently on all metrics. With a further refinement in the Focus stage by localizing clips in the selected top 20 candidate videos, all recall rates with different settings of K are boosted considerably. For example, Recall@1 is improved by about 20%, and Recall@5 is improved by about 8%.

We also evaluate our framework on the modified LSMDC dataset. From the results shown in Table 2, we observe similar trends, but more obvious.

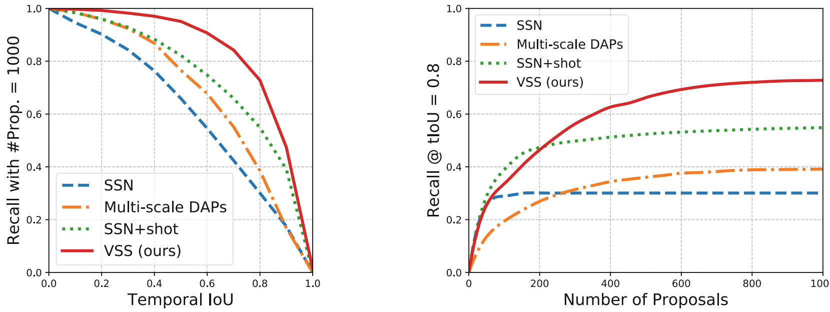
Table 1. Results for whole video retrieval on ActivityNet Captions.

	R@1	R@5	R@10	R@50	MedR
Random	0.02	0.10	0.20	1.02	2458
LSTM-YT [33]	0	4	-	24	102
S2VT [32]	5	14	-	32	78
Krishna *et al.* [17]	14	32	-	65	34
VSE (Find)	11.69	34.66	50.03	85.66	10
Ours (Find + refine in Top 20)	**14.11**	37.12	52.13	-	10
Ours (Find + refine in Top 100)	14.05	**37.40**	**52.94**	**86.72**	**9**

Table 2. Results for whole video retrieval on modified LSMDC dataset.

	R@1	R@5	R@10	R@50	MedR
Random	0.20	1.02	2.04	10.22	244
VSE (Find)	2.66	10.63	16.36	52.97	45
Ours (Find + refine in Top 20)	**3.89**	**13.70**	20.04	-	45
Ours (Find + refine in Top 70)	**3.89**	13.50	**20.25**	**56.65**	**40**

Compared to VSE, our method improves Recall@1 by about 46% (from 2.66 to 3.89) and Recall@5 by about 29% (from 10.63 to 13.70).



(a) Recall at different tIoU thresholds (b) Recall@0.8 *w.r.t.* proposal numbers

Fig. 4. Comparison of different proposal generation methods on ActivityNet Captions.

5.4 Proposal Generation and Clip Localization

We evaluate the performance of our proposal generation method, *visual semantic similarity* (VSS), in comparison with previous methods on ActivityNet Captions dataset. The performance is measured in terms of the recall rate at different tIoU thresholds. From the results shown in Fig. 4(a), we can see that our method outperforms all the other methods consistently across all tIoU thresholds. Particularly, with the tIoU threshold set to 0.5, our method can achieve a high recall 95.09% with 1000 proposals, significantly outperforming SSN+shot, a state-of-the-art method for video clip proposal, which achieves recall 84.35% with 1000 proposals. The performance gain is primarily thanks to our design that employs semantic significance instead of actionness in proposal rating.

Figure 4(b) shows that when we increase the number of proposals, the recall improves consistently and significantly. This suggests that our method tends to produce new proposals covering different temporal regions.

Table 3. Comparison of clip localization performance for different proposal methods.

ActivityNet, clip localization Recall@tIoU			
	Recall@0.3	Recall@0.5	Recall@0.7
SSN [43]	15.85	7.33	3.20
SSN [43]+shot [1]	16.71	8.74	4.30
Ours (VSS)	**28.52**	**13.46**	**5.21**

Furthermore, we compare the quality of temporal proposals generated by different methods in the task of clip localization. The performance is measured by the recall rate with different tIoU thresholds. Table 3 shows the results. Again, our proposal generation method outperforms others by a large margin.

5.5 Ablation Studies

Different Language Representations. We compare the performance of different ways to represent text on ActivityNet Captions dataset. The first two rows in Table 4 show the filtering effect of TF-IDF. The bottom two rows demonstrate that using a better word aggregation method will lead to a performance promotion, as Fisher vector [10] models a distribution over words.

Table 4. Different word representations for video retrieval on ActivityNet caption.

		R@1	R@5	R@10	R@50	MedR
BoW with tf-idf	(Find)	11.69	34.66	50.03	85.66	10
	(Find + refine in Top 100)	14.05	37.40	52.94	86.72	9
BoW without tf-idf	(Find)	11.57	33.03	49.89	85.66	11
	(Find + refine in Top 100)	13.46	36.67	52.09	86.26	9
word2vec	(Find)	9.05	27.96	42.95	81.55	14
	(Find + refine in Top 100)	10.92	32.38	46.55	82.06	12
word2vec + Fisher Vec	(Find)	11.80	34.35	50.07	85.93	10
	(Find + refine in Top 100)	13.75	37.93	53.41	86.30	9

Choice of K in Video Selection. Here, K is the number of videos retained in the initial Find stage. We compare the influence of K on the final retrieval performance, with the results reported in Table 5. The results demonstrate that the Focus stage can significantly improve the retrieval results. Generally, increasing K can lead to better performance. However, on ActivityNet Captions, as K goes beyond 20, the performance gradually saturates. Note that when K is set to a very large number ($K = 1000$), we can get almost 100% recall in Find stage. But the results are close to $K = 100$ with high computational cost.

Table 5. Retrieval performance on ActivityNet Captions with different settings of K.

	Recall@1	Recall@5	Recall@10	Recall@15	Recall@20	Recall@50
No refinement	11.69	34.66	50.03	59.90	67.34	85.66
$K = 10$	13.93	36.65	-	-	-	-
$K = 20$	**14.11**	37.12	52.13	61.62	-	-
$K = 50$	14.05	37.40	52.90	**63.29**	70.53	-
$K = 100$	14.05	37.40	52.94	63.27	**70.75**	**86.72**
$K = 1000$	14.01	**37.44**	**53.06**	63.11	70.34	86.62

Table 6. The influence caused by feature refinement under the task of clip localization.

Dataset	ActivityNet Captions			Modified LSMDC		
Clip localization Recall@tIoU	R@0.3	R@0.5	R@0.7	R@0.3	R@0.5	R@0.7
VSS (not refined feature)	27.04	12.74	4.72	5.00	2.48	0.75
VSS (refined feature)	**28.52**	**13.46**	**5.21**	**5.25**	**2.49**	**0.86**

Table 7. Comparison of the performance between different settings of the bipartite matching algorithm in the focus stage.

$u_{max} = 1$		$u_{max} = 2$		$u_{max} = 3$	
Recall@1	Recall@5	Recall@1	Recall@5	Recall@1	Recall@5
13.87	36.61	**13.93**	**36.65**	13.75	36.59

Feature Refinement. Recall that the embedded features in the Find stage can be further refined during the Focus stage. Here, we compare the performance in the task of clip localization, with or without feature refinement. The performance is measured by the recall rate of clip localization at different tIoU thresholds. The results in Table 6 show that the feature refinement in the Focus stage leads to more favorable features, which could better capture the semantic relevance across modalities.

Bipartite Matching. We try different settings for bipartite matching in the Focus stage, by varying u_{max}, the maximum number of clips allowed to be

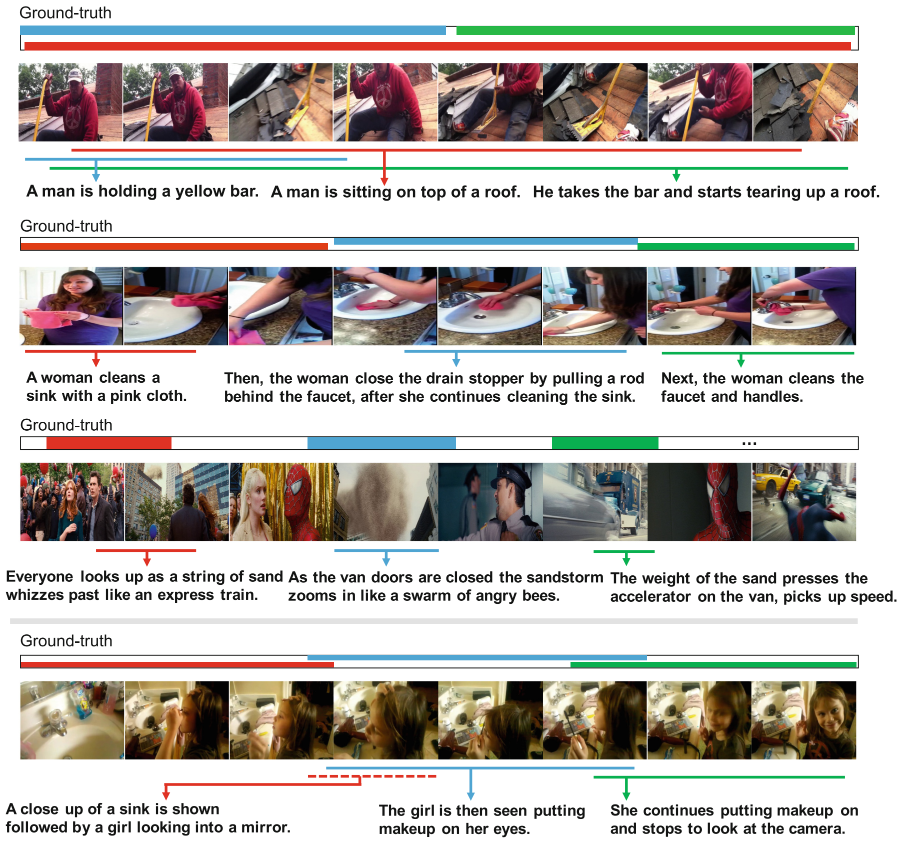

Ground-truth
A man is holding a yellow bar.
A man is sitting on top of a roof.
He takes the bar and starts tearing up a roof.
Ground-truth
A woman cleans a sink with a pink cloth.
Then, the woman close the drain stopper by pulling a rod behind the faucet, after she continues cleaning the sink.
Next, the woman cleans the faucet and handles.
Ground-truth
...
Everyone looks up as a string of sand whizzes past like an express train.
As the van doors are closed the sandstorm zooms in like a swarm of angry bees.
The weight of the sand presses the accelerator on the van, picks up speed.
Ground-truth
A close up of a sink is shown followed by a girl looking into a mirror.
The girl is then seen putting makeup on her eyes.
She continues putting makeup on and stops to look at the camera.

Fig. 5. Qualitative results of video retrieval and clip localization on ActivityNet Captions and modified LSMDC datasets. For every video with several representative frames, the ground-truth video clip is denoted in colored bars above. The localized clips associated with the query sentences are illustrated below each video. (Color figure online)

matched to a sentence. Table 7 shows that slightly increasing u_{max} can moderately improve the retrieval results, as it makes the matching process more resilient to outliers. However, the performance gain diminishes when u_{max} is too large due to the confusion brought by the increased matching clips. We observe that on ActivityNet Captions, the bipartite matching achieves the best performance when u_{max} is set to 2, and this setting is also adopted in our experiments.

5.6 Qualitative Results

We present the qualitative results of the joint video retrieval and clip localization on both ActivityNet Captions and modified LSMDC datasets in Fig. 5. We visualize three successful cases plus one failed case. We can see that in the above three examples, the clips are accurately localized and semantically associated with the query sentences. In the failed case, the first clip is wrongly localized. It reveals that although being able to capture information about objects and the static scenes, our method sometimes ignores complex relations, *e.g.* the phrase *"followed by"* in the first query sentence. More qualitative results are provided in the supplemental materials.

6 Conclusions

In this paper, we presented a two-level structured formulation to exploit both the top-level and part-level associations between paragraphs and videos. Based upon this hierarchical formulation, we propose a two-stage *Find and Focus* framework to jointly retrieve the whole videos and localize events therein with natural language queries. Our experiments show the mutual benefits between the two stages. In particular, the top-level retrieval in the Find stage helps to alleviate the burden of clip localization; while the clip localization in the Focus stage refines the retrieval results. On both ActivtyNet Captions and the modified LSMDC, the proposed method outperforms VSE and other representative methods.

Acknowledgments. This work is partially supported by the Big Data Collaboration Research grant from SenseTime Group (CUHK Agreement No. TS1610626), the Early Career Scheme (ECS) of Hong Kong (No. 24204215), and International Partnership Program of Chinese Academy of Sciences (172644KYSB20160033).

References

1. Apostolidis, E., Mezaris, V.: Fast shot segmentation combining global and local visual descriptors. In: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 6583–6587. IEEE (2014)
2. Aytar, Y., Shah, M., Luo, J.: Utilizing semantic word similarity measures for video retrieval. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1–8. IEEE (2008)
3. Bojanowski, P., et al.: Weakly-supervised alignment of video with text. In: IEEE International Conference on Computer Vision (ICCV), pp. 4462–4470 (2015)
4. Chen, K., Song, H., Loy, C.C., Lin, D.: Discover and learn new objects from documentaries. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1111–1120. IEEE (2017)
5. Dalton, J., Allan, J., Mirajkar, P.: Zero-shot video retrieval using content and concepts. In: the 22nd ACM International Conference on Information and Knowledge Management (CIKM), pp. 1857–1860. ACM (2013)
6. Faghri, F., Fleet, D.J., Kiros, J.R., Fidler, S.: VSE++: improved visual-semantic embeddings. arXiv preprint arXiv:1707.05612 (2017)
7. Frome, A., Corrado, G.S., Shlens, J., Bengio, S., Dean, J., Mikolov, T., et al.: Devise: a deep visual-semantic embedding model. In: Advances in Neural Information Processing Systems (NIPS), pp. 2121–2129 (2013)
8. Gaidon, A., Harchaoui, Z., Schmid, C.: Temporal localization of actions with actoms. IEEE Trans. Pattern Anal. Mach. Intell. **35**(11), 2782–2795 (2013)

9. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 770–778 (2016)
10. Jain, M., Van Gemert, J., Jégou, H., Bouthemy, P., Snoek, C.: Action localization with tubelets from motion. In: IEEE International Conference on Computer Vision and Pattern Recognition (CVPR) (2014)
11. Johnson, J., et al.: Image retrieval using scene graphs. In: IEEE Conference on Computer vision and Pattern Recognition (CVPR), pp. 3668–3678 (2015)
12. Jouili, S., Tabbone, S.: Hypergraph-based image retrieval for graph-based representation. Pattern Recognit. **45**(11), 4054–4068 (2012)
13. Karpathy, A., Fei-Fei, L.: Deep visual-semantic alignments for generating image descriptions. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3128–3137 (2015)
14. Karpathy, A., Joulin, A., Fei-Fei, L.: Deep fragment embeddings for bidirectional image sentence mapping. In: Advances in Neural Information Processing Systems (NIPS), pp. 1889–1897 (2014)
15. Kingma, D.P., Ba, J.: Adam: a method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
16. Kiros, R., Salakhutdinov, R., Zemel, R.S.: Unifying visual-semantic embeddings with multimodal neural language models. arXiv preprint arXiv:1411.2539 (2014)
17. Krishna, R., Hata, K., Ren, F., Fei-Fei, L., Niebles, J.C.: Dense-captioning events in videos. In: IEEE International Conference on Computer Vision (ICCV) (2017)
18. Lin, D., Fidler, S., Kong, C., Urtasun, R.: Visual semantic search: retrieving videos via complex textual queries. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2657–2664 (2014)
19. Liu, W., Mei, T., Zhang, Y., Che, C., Luo, J.: Multi-task deep visual-semantic embedding for video thumbnail selection. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3707–3715 (2015)
20. Otani, M., Nakashima, Y., Rahtu, E., Heikkilä, J., Yokoya, N.: Learning joint representations of videos and sentences with web image search. In: Hua, G., Jégou, H. (eds.) ECCV 2016. LNCS, vol. 9913, pp. 651–667. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46604-0_46
21. Plummer, B.A., Brown, M., Lazebnik, S.: Enhancing video summarization via vision-language embedding. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2017)
22. Ren, M., Kiros, R., Zemel, R.: Image question answering: a visual semantic embedding model and a new dataset. Adv. Neural Inf. Process. Systems (NIPS) **1**(2), 5 (2015)
23. Rohrbach, A., et al.: Movie description. Int. J. Comput. Vis. **123**(1), 94–120 (2017)
24. Sharghi, A., Gong, B., Shah, M.: Query-focused extractive video summarization. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9912, pp. 3–19. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46484-8_1
25. Shou, Z., Wang, D., Chang, S.F.: Temporal action localization in untrimmed videos via multi-stage CNNs. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1049–1058 (2016)
26. Smoliar, S.W., Zhang, H.: Content based video indexing and retrieval. IEEE Multimed. **1**(2), 62–72 (1994)
27. Snoek, C.G., Worring, M.: Concept-based video retrieval. Found. Trends Inf. Retrieval **2**(4), 215–322 (2008)

28. Tang, K., Yao, B., Fei-Fei, L., Koller, D.: Combining the right features for complex event recognition. In: IEEE International Conference on Computer Vision (ICCV), pp. 2696–2703. IEEE (2013)
29. Tapaswi, M., Bäuml, M., Stiefelhagen, R.: Aligning plot synopses to videos for story-based retrieval. Int. J. Multimed. Inf. Retrieval **4**(1), 3–16 (2015)
30. Torabi, A., Tandon, N., Sigal, L.: Learning language-visual embedding for movie understanding with natural-language. arXiv preprint arXiv:1609.08124 (2016)
31. Vendrov, I., Kiros, R., Fidler, S., Urtasun, R.: Order-embeddings of images and language. In: International Conference on Representation Learning (ICLR) (2016)
32. Venugopalan, S., Rohrbach, M., Donahue, J., Mooney, R., Darrell, T., Saenko, K.: Sequence to sequence-video to text. In: IEEE International Conference on Computer Vision (ICCV), pp. 4534–4542 (2015)
33. Venugopalan, S., Xu, H., Donahue, J., Rohrbach, M., Mooney, R., Saenko, K.: Translating videos to natural language using deep recurrent neural networks. arXiv preprint arXiv:1412.4729 (2014)
34. Wang, D., Li, X., Li, J., Zhang, B.: The importance of query-concept-mapping for automatic video retrieval. In: the 15th ACM International Conference on Multimedia, pp. 285–288. ACM (2007)
35. Wang, L., et al.: Temporal segment networks: towards good practices for deep action recognition. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9912, pp. 20–36. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46484-8_2
36. Wang, L., Li, Y., Lazebnik, S.: Learning deep structure-preserving image-text embeddings. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 5005–5013 (2016)
37. Wu, B., Lang, B., Liu, Y.: GKSH: graph based image retrieval using supervised kernel hashing. In: International Conference on Internet Multimedia Computing and Service, pp. 88–93. ACM (2016)
38. Xu, R., Xiong, C., Chen, W., Corso, J.J.: Jointly modeling deep video and compositional text to bridge vision and language in a unified framework. In: AAAI Conference on Artificial Intelligence (AAAI), vol. 5, p. 6 (2015)
39. Yao, L., et al.: Describing videos by exploiting temporal structure. In: IEEE International Conference on Computer Vision (ICCV), pp. 4507–4515 (2015)
40. Yeung, S., Russakovsky, O., Mori, G., Fei-Fei, L.: End-to-end learning of action detection from frame glimpses in videos. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2678–2687 (2016)
41. Yu, Y., Ko, H., Choi, J., Kim, G.: End-to-end concept word detection for video captioning, retrieval, and question answering. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2017)
42. Zhang, H.J., Wu, J., Zhong, D., Smoliar, S.W.: An integrated system for content-based video retrieval and browsing. Pattern Recognit. **30**(4), 643–658 (1997)
43. Zhao, Y., Xiong, Y., Wang, L., Wu, Z., Tang, X., Lin, D.: Temporal action detection with structured segment networks. In: IEEE International Conference on Computer Vision (ICCV), vol. 8 (2017)
44. Zhu, Y., et al.: Aligning books and movies: towards story-like visual explanations by watching movies and reading books. In: IEEE International Conference on Computer Vision (ICCV), pp. 19–27 (2015)