Multilevel Language and Vision Integration for Text-to-Clip Retrieval

Huijuan Xu,[1] **Kun He,**[1] **Bryan A. Plummer,**[1] **Leonid Sigal,**[2] **Stan Sclaroff,**[1] **Kate Saenko**[1]

[1]Boston University, [2]University of British Columbia

hxu@bu.edu, hekun@bu.edu, bplum@bu.edu, lsigal@cs.ubc.ca, sclaroff@bu.edu, saenko@bu.edu

Abstract

We address the problem of text-based activity retrieval in video. Given a sentence describing an activity, our task is to retrieve matching clips from an untrimmed video. To capture the inherent structures present in both text and video, we introduce a multilevel model that integrates vision and language features earlier and more tightly than prior work. First, we inject text features early on when generating clip proposals, to help eliminate unlikely clips and thus speed up processing and boost performance. Second, to learn a fine-grained similarity metric for retrieval, we use visual features to modulate the processing of query sentences at the word level in a recurrent neural network. A multi-task loss is also employed by adding query re-generation as an auxiliary task. Our approach significantly outperforms prior work on two challenging benchmarks: Charades-STA and ActivityNet Captions.

Introduction

Temporal localization of events or activities of interest is a key problem in computer vision. Recently there has been increased interest in specifying the queries using natural language rather than only supporting a predefined set of actions or events (Gao et al. 2017; Hendricks et al. 2017). In this paper, we focus on the task of retrieving temporal segments in untrimmed video through natural language queries, or simply, *text-to-clip*. Solving this task requires understanding the nuances of natural language and video contents. For example, in Figure 1, although the two queries talk about the same objects, their verbs make the key difference, and the temporal ordering of video frames can give an important cue.

Existing methods for solving cross-modal retrieval tasks typically learn embedding functions to project data from different modalities into a common vector embedding space. In this common space retrieval is performed using standard similarity metrics, such as Euclidean distance. However, for *text-to-clip*, such wholistic representations of sentences and video clips make it difficult to leverage fine-grained structures, such as the ordering of words and frames. Also, the embeddings are usually independent of each other, making it impossible to use input from one modality to modulate the processing of another.
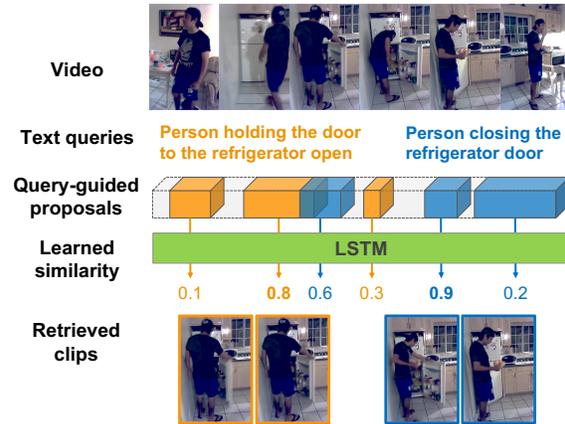
Figure 1: *Text-to-clip* is a complicated task that can involve objects, their interactions, and activities. In this example, the model needs to understand the objects being referred to (person and refrigerator), disambiguate between the verbs (holding open *vs.* closing), and perform temporal localization based on the movement of the refrigerator door. For this task, we propose a *multilevel* model to tightly integrate language and vision features in its retrieval pipeline: 1) query-guided video clip proposals, 2) learned similarity measure with a recurrent LSTM.

To better model the structured information present in both modalities, we propose a novel *multilevel* approach for integrating vision and language features for the text-to-clip task. In the first level, we inject language features at the temporal segment proposal stage. Inspired by attention mechanisms in Visual Question Answering (VQA) (Anderson et al. 2018; Lu et al. 2016; Shih, Singh, and Hoiem 2016; Xu and Saenko 2016), we re-weight the video features used to propose temporal segments by their similarity to the encoding of the input query. This enables our model to select clips that are more relevant to queries as candidates for further processing. The video feature weights can be pooled over a set of queries which helps to reduce computational costs when performing multiple queries on a single video.

The second level of integration happens when computing the similarity between the text query and video clips. We learn a Long Short-Term Memory (LSTM) model that pro-

cesses the query sentence word-by-word, conditioned on the visual feature embedding of a candidate clip in each step, and produces a nonlinear similarity score. This can be seen as an *early fusion* of vision and language, as opposed to the *late fusion* in vector embedding approaches. Also, our model has the freedom to associate each word in the query sentence with a potentially different part of the visual feature when computing the similarity, which is not possible in independent vector embeddings. Additionally, we train our model with a multi-task loss, adding *clip-to-text* as an auxiliary task, which is also known as dense video captioning (Krishna et al. 2017). We show that learning a shared representation for both tasks improves retrieval performance. An overview of our approach is provided in Figure 1.

Our approach differs from prior work in two important ways. First, our model learns to propose temporal segments conditioned on the query, unlike exiting solutions for this task that employ sliding windows or hand-crafted heuristics (Hendricks et al. 2017; Gao et al. 2017) that do not consider the query at all. We generalize the R-C3D model (Xu, Das, and Saenko 2017) to create query-specific proposals. Second, we improve upon vector embedding approaches that pool the hidden states of a recurrent neural network to obtain sentence embeddings (Hendricks et al. 2017; Gao et al. 2017; Aytar, Vondrick, and Torralba 2017). These models would have to anticipate what words are important without having access to the visual data. Our approach addresses this drawback by integrating visual features while processing sentence queries at the word level.

To summarize our contributions, for the text-to-clip retrieval task, in this paper we

- incorporate query embeddings into the segment proposal network to generate query-guided proposals;

- take an early fusion approach and learn an LSTM to model fine-grained similarity between query sentences and video clips;

- leverage captioning as an auxiliary task to learn better shared feature representations.

We conduct extensive evaluation and ablation studies on two challenging benchmarks: Charades-STA (Gao et al. 2017) and ActivityNet Captions (Krishna et al. 2017). Our full model achieves state-of-the-art performance. Code is released for public use[1].

Related Work

Temporal Activity Detection: Temporal activity detection aims at predicting the start and end times of the activities within untrimmed videos. Early approaches (Shou, Wang, and Chang 2016) use sliding windows to generate segments and subsequently classify them, which is computationally inefficient and constrains the granularity of detection. Later in (Singh et al. 2016; Serena Yeung afinally, Mori, and Fei-Fei 2016) temporal localization is obtained by modeling the evolution of activities using Recurrent Neural Networks (RNNs) and predicting activity labels or activity segments at

each time step. R-C3D (Xu, Das, and Saenko 2017) adapts the proposal and classification pipeline from object detection (Ren et al. 2015) to perform activity detection using 3D convolutions (Tran et al. 2015) and 3D Region of Interest pooling. SSAD (Lin, Zhao, and Shou 2017) performs single-shot temporal activity detection following the one-stage object detection method SSD (Liu et al. 2016).

Instead of treating activities as distinct classes and using a discrete and fixed vocabulary of class labels, language queries in video language localization task can express more semantic meaning. In this paper, we use a proposal-based pipeline to solve the video language localization task, and adopt the proposal generation technique of R-C3D.

Vision and Language: In this paper our task is to locate the visual events that match a query sentence in a video. There are two main types of approaches to solve such cross-modal retrieval tasks: early fusion and late fusion. The late fusion approach embeds different modalities into a common embedding space, and then measure their similarity. These approaches are not restricted to vision and language, but can be applied across modalities such as image, video, text, and sound (Arandjelović and Zisserman 2017; Aytar, Vondrick, and Torralba 2017; Vendrov et al. 2016). Early fusion approaches combine the features from each modality at an earlier stage (Ma et al. 2015; Wang et al. 2018; Yu et al. 2017) and predicts similarity scores directly based on the fused feature representation. (de Vries et al. 2017) argues against the dominant late-fusion pipeline, where linguistic inputs are mostly processed independently, and shows that modulating visual representations with language at earlier levels improves visual question answering.

For the text-to-clip task considered in this paper, existing models (Gao et al. 2017; Hendricks et al. 2017) perform late fusion and embed entire query sentences to vectors. However, as we argued in the introduction, this tends to lose important information about fine-grained structures. We propose a novel model that performs early fusion of the video and query features, combining them at the word level, and we compare it with the sentence-level fusion approach.

Other typical vision-language tasks include image/video captioning (Donahue et al. 2015; Venugopalan et al. 2015b; 2015a; Vinyals et al. 2015; Yao et al. 2015; Xu et al. 2015) and visual question answering (VQA) (Xiong, Merity, and Socher 2016; Yang et al. 2016). We note that these tasks are rarely isolated and often influence each other. For example, image captioning can be solved as a retrieval task (Fang et al. 2015). Also, there is recent research that suggests that VQA can be leveraged to benefit the image-caption retrieval task (Lin and Parikh 2016). Our proposed multi-task formulation, which uses captioning as an auxiliary task, is partly motivated by these observations.

Localization-based Cross-modal Tasks: Several vision-language tasks also share the need for a localization component. In the dense captioning task, models need to localize interesting events in images (Johnson, Karpathy, and Fei-Fei 2016) or videos (Krishna et al. 2017; Xu et al. 2019) and provide textual descriptions. Recently, the task of grounding text in images (Hu et al. 2016; Rohrbach et al. 2016) has

---

[1]https://github.com/VisionLearningGroup/Text-to-Clip_Retrieval

been extended into videos, which introduces the task of retrieving video segments using language queries (Hendricks et al. 2017; Gao et al. 2017). These visual grounding approaches have included models which reconstruct the text query (*e.g.* (Javed Syed Ashar and Vineet 2018; Rohrbach et al. 2016)), which we also take advantage of. We note that the localization mechanisms in (Hendricks et al. 2017; Gao et al. 2017) are either inefficient (sliding-window based) or inflexible (hard-coded). In contrast with these approaches, we adopt learned segment proposals in our pipeline.

Approach

We propose a novel approach for temporal activity localization and retrieval based on input language queries, or the text-to-clip task. Our key idea is to integrate language and vision more closely before computing a match, using an early fusion scheme, query-specific proposals, and a multi-task formulation that re-generates the caption.

We first define the cross-modal retrieval problem we are solving. Given an untrimmed video V and a sentence query S, the goal is to retrieve a temporal segment (clip) R in V that best corresponds to S. In other words, we learn a mapping $\mathcal{F}_{\text{RET}} : (V, S) \mapsto R$. At training time, we are given a set of annotated videos $\{V_1, V_2, \ldots, V_N\}$. For each video V_i, its annotation is a set of matching sentence-clip pairs $A_i = \{(S_{ij}, R_{ij})\}_{j=1}^{n_i}$, where S_{ij} is a sentence, and clip $R_{ij} = (t_{ij}^0, t_{ij}^1)$ is represented as a pair of timestamps that define its start and end. We tackle the retrieval problem through learning a similarity score $\sigma(S, R)$ that measures how well S and R match each other. At test time, given V and S, the retrieval problem is formulated as

$$R^* = \arg\max_{R \in V} \sigma(R, S). \tag{1}$$

The remainder of this section is organized as follows. First, we introduce our query-guided temporal segment proposal network. Then, we detail how we learn a fine-grained similarity model for retrieval. Finally, we describe the multi-task loss function which combines the retrieval loss with an auxiliary captioning loss.

Query-Guided Segment Proposal Network

For unconstrained localization in videos, it is important to generate variable-length candidate temporal segments for further processing. Instead of using handcrafted heuristics or computationally expensive multiscale sliding windows, we employ a learned segment proposal network (SPN), similar to the one used in R-C3D (Xu, Das, and Saenko 2017) for action localization. The SPN first encodes all frames in an input video using a 3D convolutional network (C3D) (Tran et al. 2015). Then, variable-length segment proposals are obtained by predicting relative offsets to a set of predefined anchor segments. The proposal features are generated by 3D Region of Interest Pooling.

We note that the original SPN from R-C3D aims at finding "anything interesting" for unconstrained action localization. However, in the text-to-clip task where a query is specified, the search space can be further reduced by only generating proposals that are relevant to the query. We thus
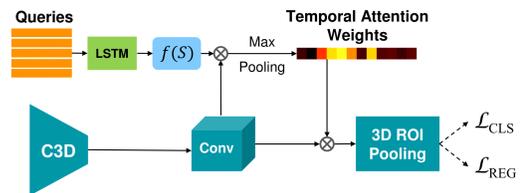

Figure 2: The structure of the query-guided Segment Proposal Network. Query embeddings are used to derive attention weights and re-weight the video convolutional features before generating proposals. The SPN is trained with a combination of classification loss $\mathcal{L}_{\text{CLS}}$ and regression loss $\mathcal{L}_{\text{REG}}$.

develop a *query-guided* SPN. The basic idea is to use the feature representation of the query to modulate the SPN and attend to the relevant temporal regions, in a way that is similar to attention mechanisms in Visual Question Answering (VQA) (Lu et al. 2016; Xu and Saenko 2016; Shih, Singh, and Hoiem 2016).

Figure 2 shows the structure of the query-guided SPN. The original SPN is shown at the bottom. Each query sentence S is embedded into a feature vector $f(S)$ by pooling the hidden states of a sentence embedding LSTM (described in the next subsection). Then, for each temporal location, an attention weight is computed by taking the inner product of the video features and $f(S)$, and passing through the $\tanh$ activation. The attention weight at each temporal location is multiplied with the corresponding video features across all channels. When there are multiple queries, we max-pool the weights over the query dimension.

Early Fusion Retrieval Model

The output from the segment proposal network is a set of temporal segments likely to contain the relevant activity, along with their pooled C3D features. We next need a retrieval model to find the segment that best matches a query.

As shown in Fig. 3, the pooled C3D features of a clip, along with the query sentence, are fed as input to a two-layer LSTM. The first layer of the LSTM processes the words in the sentence. In the second layer, the visual feature embedding is used as input at each step, along with hidden states from the sentence embedding LSTM. The final hidden state is passed through additional layers to predict a scalar similarity value. We note that while our approach does increase the number of learnable parameters in the model, it also brings additional structure into the similarity metric. This comes from each word in the sentence now being able to interact with the visual features, enabling the model to learn a potentially different way to associate each word and the visual features. We do not explicitly use attention mechanisms to enforce such behavior, but instead let the LSTM learn in a data-driven manner. Our retrieval model is an *early fusion* model, where the processing of visual features and language features are intertwined rather than isolated.

To train the retrieval model, we use a triplet-based retrieval loss, also called pairwise ranking loss (Hüllermeier et al. 2008), which has shown good performance in
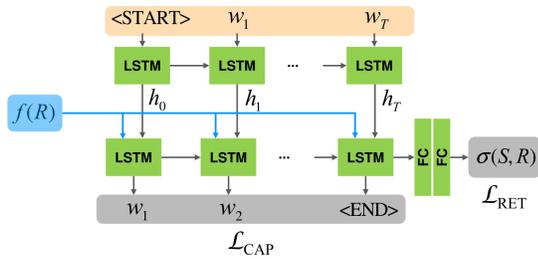
Figure 3: Our early fusion retrieval model with multi-task loss, instantiated as a two-layer LSTM. The first layer embeds the sentence query S, while the second layer takes both h_0 and the visual feature of a clip $f(R)$ as (separate) inputs, and predicts a nonlinear similarity score $\sigma(S, R)$ that is supervised by the retrieval loss $\mathcal{L}_{\text{RET}}$. Additionally, we add a captioning loss $\mathcal{L}_{\text{CAP}}$ to enforce the LSTM to re-generate the query sentence, resulting in improved retrieval performance.

metric learning tasks (Hoffer and Ailon 2015; Schroff, Kalenichenko, and Philbin 2015). Specifically, we take triplets of the form (S, R, R') where (S, R) is a matching sentence-clip pair, and R' is a negative clip that does not match S. Note that R' can either come from the same video as R with a sufficiently low overlap, or from a different video. The loss encourages the similarity score between the matching pair, $\sigma(S, R)$, to be greater than $\sigma(S, R')$ by some margin $\eta > 0$:

$$\mathcal{L}_{\text{RET}} = \sum_{(S,R,R')} \max\{0, \eta + \sigma(S, R') - \sigma(S, R)\}. \quad (2)$$

In our model, $\sigma(S, R)$ is directly predicted by the LSTM rather than using some generic measure, like cosine similarity, as done in prior work (*e.g.* (Hendricks et al. 2017)).

Multi-Task Loss

After defining the retrieval model, we now seek to gain additional benefit from training on closely related tasks. Specifically, we add a captioning loss which can act as a verification step for our model, *i.e.* we should be able to re-generate the query sentence from the retrieved video clip. Captioning has also proven to improve performance on image-based multimodal retrieval tasks (Rohrbach et al. 2016). Moreover, it is observed (Ramanishka et al. 2017) that captioning models can implicitly learn features and attention mechanisms to associate spatiotemporal regions to words in the captions. As for implementation, the paired sentence-clip annotation format in the text-to-clip task allows us to easily add captioning capabilities to our LSTM model.

As shown in Figure 3, we require the second layer of our LSTM to re-generate the input query sentence, conditioned on the proposal's visual features $f(R)$ at each step. When generating word w_t at step t, the hidden state from the previous step in the sentence embedding LSTM, $h_{t-1}^{(1)}$, is used as input. We use a standard captioning loss that maximizes the normalized log likelihood of the words generated at all T unrolled time steps, over all K ground truth matching sentence-clip pairs:

$$\mathcal{L}_{\text{CAP}} = -\frac{1}{KT} \sum_{k=1}^{K} \sum_{t=1}^{T_k} \log P(w_t^k | f(R), h_{t-1}^{(2)}, w_1^k, ..., w_{t-1}^k). \quad (3)$$

Implementation Details

Our multi-task model uses weighting parameter λ to optimize a combination of retrieval loss and captioning loss:

$$\mathcal{L} = \mathcal{L}_{\text{RET}} + \lambda \mathcal{L}_{\text{CAP}}. \quad (4)$$

We choose $\lambda = 0.5$ through cross-validation. The margin parameter η is set to 0.2 in the retrieval loss $\mathcal{L}_{\text{RET}}$. During training, each minibatch contains 32 matching sentence-clip pairs sampled from the training set, which are then used to construct triplets. We use the Adam optimizer (Kingma and Ba 2014) with learning rate 0.001 and early stopping on the validation set, for 30 epochs in total.

For the sentence embedding LSTM (first layer), we use `word2vec` (Mikolov et al. 2013) as the input word representation. The word embeddings are 300-dimensional, and trained from scratch on each dataset. The hidden state size of the LSTM is set to 512. The size of common embedding space in the late fusion retrieval model is 1024.

The similarity score for a sentence-clip pair is produced by taking the hidden state corresponding to the last word in the second layer of the LSTM, and passing it through two fully-connected (FC) layers followed by a sigmoid activation to produce a scalar value σ, as shown in Fig. 3. The two FC layers reduce the dimensionality from 512 to 64 to 1.

At test time, retrieving clips in untrimmed videos involves searching over all possible segments. Candidate proposal segments generated from the proposal network are filtered by non-maximum suppression with threshold 0.7, and the top 100 proposals in each video are used for retrieval.

Experiments

We evaluate our proposed models on one recent dataset designed for the text-to-clip retrieval task, Charades-STA (Gao et al. 2017), and one dataset designed for the dense video caption task which has the data annotations required by the text-to-clip retrieval task, ActivityNet Captions dataset (Krishna et al. 2017). We compare three versions of our model and two baselines:

- Random: a trivial baseline that randomly selects among among candidate clips.

- VE: vector embedding approach that separately embeds the query sentence and video clip to vectors.

- LSTM: our early fusion model that predicts query-clip similarity from the LSTM.

- LSTM+QSPN: LSTM with query-guided segment proposal network, as opposed to query-agnostic proposals.

- LSTM+QSPN+Cap: our full model with the addition of captioning loss.

We follow the evaluation setup in (Gao et al. 2017), which is adapted from a similar task in the image domain, namely, natural language object retrieval (Hu et al. 2016).

Methods	tIoU=0.3			tIoU=0.5			tIoU=0.7		
	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
Random (Gao et al. 2017)	–	–	–	8.5	37.1	–	3.0	14.1	–
CTRL(reg-np) (Gao et al. 2017)	–	–	–	23.6	58.9	–	8.9	29.5	–
VE	43.9	83.5	89.7	26.3	63.9	78.2	10.9	35.6	50.5
LSTM	51.6	95.5	99.0	32.8	76.3	92.5	14.0	43.2	60.7
LSTM+Cap	52.3	95.3	**99.2**	34.4	77.0	92.5	15.6	44.9	61.4
LSTM+QSPN	54.1	**95.8**	**99.2**	35.3	77.8	93.5	15.2	44.6	61.9
LSTM+QSPN+Cap	**54.7**	95.6	**99.2**	**35.6**	**79.4**	**93.9**	**15.8**	**45.4**	**62.2**

Table 1: Results on the Charades-STA dataset (Gao et al. 2017). $R@K$ stands for Recall@K. Our early fusion retrieval model LSTM significantly outperforms baselines, while the multi-task and query-guided proposals in model LSTM+QSPN+Cap further improve results.

queries: [start:end] sentence	Expected	LSTM+Cap	LSTM	VE
video ID: EEVD3 1. [2.0:7.9] a person is holding the door to the refrigerator open. 2. [11.4:16.9] person closing the door.				
video ID: 3VT73 1. [2.3:11.6] a person sits down as they read a book. 2. [8.4:13.1] the person throws a book. 3. [10.4:15.9] person he takes his cell phone out.				
video ID: 0LHWF 1. [0.0:4.1] a person sits in a chair. 2. [2.1:16.2] person holding a book. 3. [2.7:15.0] person reading a book. 4. [2.7:15.0] person read book.				

Table 2: Visualization of similarity scores between the N input queries for a video and their N ground truth temporal segments (resulting in a $N \times N$ confusion matrix) for the models LSTM+Cap, LSTM and VE on Charades-STA dataset. Warmer colors indicate higher similarity scores. The start and end times are in seconds.

Specifically, we consider a set of temporal Intersection-Over-Union (tIoU) thresholds. For each threshold τ, we compute the Recall@K metric, defined as the fraction of sentence queries having at least one correct retrieval (having tIoU greater than τ with ground truth) in the top K retrieved video clips. Following standard practice, we use $\tau \in \{0.3, 0.5, 0.7\}$ and $K \in \{1, 5, 10\}$.

Experiments on the Charades-STA Dataset

Dataset and Setup: The Charades-STA dataset was introduced in (Gao et al. 2017) for evaluating temporal localization of events in video given natural language queries. The original Charades dataset (Sigurdsson et al. 2016) only provides a paragraph description for each video. To generate sentence-clip annotations used in the retrieval task, (Gao et al. 2017) decomposed the original video-level descriptions into shorter sub-sentences, and performed keyword matching to assign them to temporal segments in videos. The alignment annotations are further verified manually. The released annotations comprise 12,408 sentence-clip pairs for training, and 3,720 for testing.

We keep all the words that appear in the training set, resulting in a vocabulary size of 1,111. The maximum caption length is set to 10. We sample frames at 5 fps and set the number of input frames to 768, breaking arbitrary-length input videos into 768-frame chunks, and zero-padding them if necessary. To initialize our segment proposal network, we finetune a C3D model (Tran et al. 2015) pretrained on Sports-1M (Karpathy et al. 2014), with the ground truth activity segments of 157 classes in the training videos of the Charades activity detection dataset. We then extract proposal visual features and train the retrieval model.

Results: Table 1 shows the results on the text-to-clip retrieval task for Charades-STA. First, it is interesting to note that our vector embedding baseline (VE) already outperforms CTRL (reg-np), the best model in (Gao et al. 2017), by a noticeable margin. We believe there are two reasons for this. First, our segment proposal network from R-C3D model offers finer temporal granularity, and therefore provides cleaner visual feature representations compared to the sliding windows approach in CTRL. Second, we use a triplet-based loss that more effectively captures ranking con-

straints, compared to CTRL's binary classification loss.

Our LSTM model significantly outperforms the VE baseline. To explore the effectiveness of the early fusion approach for fusing cross-modal features used in the retrieval model, we visualize the similarity scores (warmer colors = higher) between ground truth queries and their corresponding ground truth segments, for three example videos from Charades-STA dataset in Table 2. The ideal result is a block-diagonal matrix shown in the "Expected" column of Table 2, which indicates that the ground truth query only has high correlation with its own ground truth segment, and can distinguish irrelevant temporal segments. In the first example, VE maps two queries about the door with opposite actions "open" and "closing" to the same segment, while LSTM and LSTM+Cap can recognize these two opposite actions on the same objects. In the second example, VE confuses between the 1st and 2nd queries that are both about "person" and "book", while LSTM starts to distinguish these two queries and LSTM+Cap distinguishes them perfectly. However, there are also some cases that VE performs slightly better as in the third example VE can distinguish "a person sits in a chair." and "person holding a book.". In general, early fusion can capture more fine-grained action details in computing similarity scores when the objects are the same, which might be the reason for the good performance of the early fusion models.
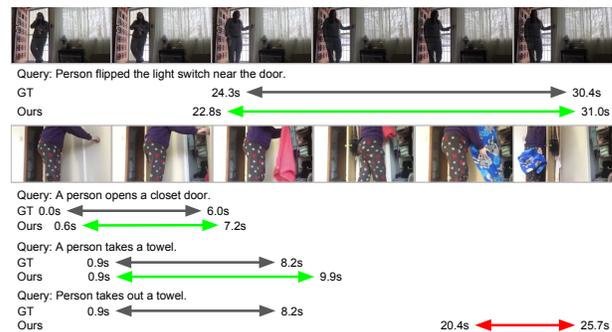
Since we share parameters between two tasks in the fusion LSTM layer, LSTM+Cap is able to further improve results on most metrics except R@5 at tIou 0.3. Further ablations of the captioning loss weight λ in Eq. 4 for training the LSTM+Cap method are shown in Table 3. As our main task is retrieval, we choose $\lambda = 0.5$ in our experiments.

When using the query-guided segment proposal network, the model LSTM+QSPN improves on all the metrics over the early fusion model LSTM. The layers used to incorporate the query in the query-guided SPN are cross-validated and shown in Supplement Material. Our full model LSTM+QSPN+Cap with both query-guided proposals and captioning supervision obtains the highest results on most metrics except a minor weakness on R@5 at tIoU 0.3.
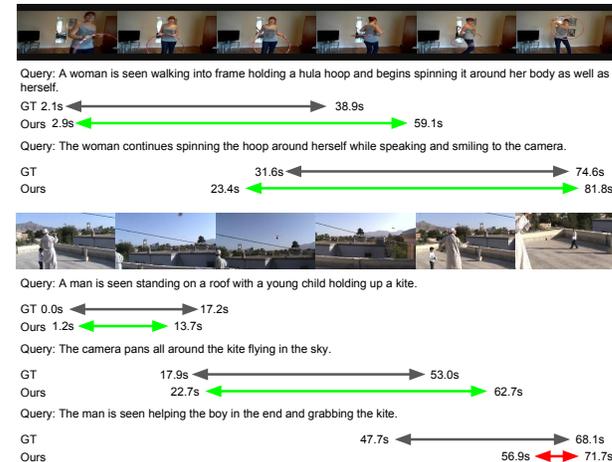
Two example videos from the Charades-STA dataset along with query localization results are shown in Figure 4(a). The correct prediction is marked as green, while the wrong one is marked as red. Please note that the prediction is in fact correct for the query *Person takes out a towel*, but is marked incorrect due to inaccurate ground truth.

Experiments on ActivityNet Captions Dataset

Dataset and Setup: The ActivityNet Captions dataset was proposed by Krishna *et al.* (Krishna et al. 2017) for the dense video captioning task which contains the temporal segment annotations and paired captions. These annotations can also be used by the text-to-clip retrieval task, where the caption sentences are used as input query sentences for each video. Each video contains at least two ground truth segments and each segment is paired with one ground truth caption. The ActivityNet Captions dataset (Krishna et al. 2017) contains around 20k videos and is split into training, validation and testing with a 50%/25%/25% ratio. We train on the training

(a) Charades-STA retrieval examples

(b) ActivityNet Captions retrieval examples

Figure 4: Qualitative visualization of example retrieval results from our full model, LSTM+QSPN+Cap, on the Charades-STA dataset (a) and the ActivityNet Captions dataset (b). Ground truth (GT) clips corresponding to queries are marked with black arrows. Correct predictions (predicted clips having temporal IoU more than 0.5 with ground truth) are marked in green, and incorrect predictions are marked in red. The start and end times are shown in seconds. Best viewed in color.

set and test on the combined validation sets in our experiments since the caption annotations in the test set are withheld for challenge purpose.

We keep the words in the training set with frequency greater than five to build a vocabulary of size 3,892, and set the maximum caption length to 30. We sample frames at 3 fps, and set the maximum number of input frames in the buffer to be 768. Again, a 3D ConvNet model (Tran et al. 2015) pretrained on the Sports-1M dataset and finetuned on ground truth activity segments of ActivityNet detection dataset is used to initialize our segment proposal network. We use the good settings from the ablation studies on the Charades-STA dataset.

Results: Results using the standard evaluation protocol are given in Table 4. Similar trends can be observed for the

Loss Weight	tIoU=0.3			tIoU=0.5			tIoU=0.7		
	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
$\lambda = 0.5$	**52.3**	**95.3**	**99.2**	**34.4**	**77.0**	**92.5**	**15.6**	**44.9**	**61.4**
$\lambda = 1$	50.8	94.5	98.1	32.5	76.1	91.2	14.1	41.9	59.2
$\lambda = 2$	50.6	94.9	98.5	33.5	76.5	91.3	14.3	43.4	60.3

Table 3: The effect of loss weight λ in the LSTM+Cap method, measured on the Charades-STA dataset. R@K stands for Recall@K. As our main task is retrieval, we consistently underweight the captioning loss with $\lambda = 0.5$ in our experiments.

Methods	tIoU=0.3			tIoU=0.5			tIoU=0.7		
	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
Random	5.6	24.8	42.8	2.5	11.3	21.6	0.8	4.0	8.1
VE	39.3	67.8	75.6	23.7	52.0	62.2	11.0	32.1	42.1
LSTM	42.2	70.5	78.0	25.7	54.5	64.0	12.6	34.1	43.7
LSTM+Cap	42.8	73.5	80.8	26.2	56.9	66.7	12.6	35.8	46.3
LSTM+QSPN	44.2	74.5	81.9	26.9	56.7	66.7	13.2	35.5	45.6
LSTM+QSPN+Cap	**45.3**	**75.7**	**83.3**	**27.7**	**59.2**	**69.3**	**13.6**	**38.3**	**49.1**

Table 4: Retrieval results on the ActivityNet Captions dataset (Krishna et al. 2017). R@K stands for Recall@K. Our full model LSTM+QSPN+Cap with query-guided segment proposal network and auxiliary captioning loss outperforms all other baseline models.

five variants of our model, as in the Charades-STA experiments. LSTM significantly outperforms the baseline VE. Also, with the assistance of the captioning loss, the multi-task model LSTM+Cap does better than LSTM. With input query sentences regulating the segment proposal network, all the metrics in the model LSTM+QSPN improve compared to using the original segment proposal network in LSTM. Our full model LSTM+QSPN+Cap with both captioning supervision and query-guided proposals gets highest results in all the metrics.

Two example retrieval results from the ActivityNet Captions dataset can be found in Figure 4(b). In the first example, our model localizes the precise moment described by the query sentences about playing hula hoop. In the second example, it also correctly identifies the events corresponding to the queries *A man is seen standing on a roof with a young child holding up a kite* and *The camera pans all around the kite flying in the sky*. However, for the third query *The man is seen helping the boy in the end and grabbing the kite*, our model directly localizes to an incorrect shot segment at the very end of the video, maybe biased by the phrase "in the end" in the input query.

Conclusion

In this paper, we address the problem of text-to-clip retrieval: temporal localization of events within videos that match a given natural language query. We introduce a multilevel feature integration model to fuse language and vision earlier and more tightly than existing methods, which are typically based on the late fusion approach of learning independent vector embeddings. We make use of a segment proposal network to filter out unlikely clips, and improve it by conditioning on the input query. We learn a two-layer LSTM to directly predict similarity scores between sentence queries and video clips, and augment its training loss by adding the captioning task.

Evaluated on two challenging datasets, our approach performs more accurately than previous methods when retrieving clips from many possible candidates in untrimmed videos. For future work, we are interested in further exploiting language features in order to modulate the extraction of visual features, similar to (de Vries et al. 2017).

References

Anderson, P.; He, X.; Buehler, C.; Teney, D.; Johnson, M.; Gould, S.; and Zhang, L. 2018. Bottom-up and top-down attention for image captioning and visual question answering. In *CVPR*.

Arandjelović, R., and Zisserman, A. 2017. Objects that sound. *arXiv preprint arXiv:1712.06651*.

Aytar, Y.; Vondrick, C.; and Torralba, A. 2017. See, hear, and read: Deep aligned representations. *arXiv preprint arXiv:1706.00932*.

de Vries, H.; Strub, F.; Mary, J.; Larochelle, H.; Pietquin, O.; and Courville, A. C. 2017. Modulating early visual processing by language. In *NIPS*.

Donahue, J.; Anne Hendricks, L.; Guadarrama, S.; Rohrbach, M.; Venugopalan, S.; Saenko, K.; and Darrell, T. 2015. Long-term recurrent convolutional networks for visual recognition and description. In *Proc. IEEE CVPR*.

Fang, H.; Gupta, S.; Iandola, F.; Srivastava, R. K.; Deng, L.; Dollár, P.; Gao, J.; He, X.; Mitchell, M.; Platt, J. C.; et al. 2015. From captions to visual concepts and back. In *Proc. IEEE CVPR*.

Gao, J.; Sun, C.; Yang, Z.; and Nevatia, R. 2017. TALL: Temporal activity localization via language query. In *Proc. IEEE ICCV*.

Hendricks, L. A.; Wang, O.; Shechtman, E.; Sivic, J.; Darrell, T.; and Russell, B. 2017. Localizing moments in video with natural language. In *Proc. IEEE ICCV*.

Hoffer, E., and Ailon, N. 2015. Deep metric learning using triplet network. In *International Workshop on Similarity-Based Pattern Recognition*.

Hu, R.; Xu, H.; Rohrbach, M.; Feng, J.; Saenko, K.; and Darrell, T. 2016. Natural language object retrieval. In *Proc. IEEE CVPR*.

Hüllermeier, E.; Fürnkranz, J.; Cheng, W.; and Brinker, K. 2008. Label ranking by learning pairwise preferences. *Artificial Intelligence* 172(16-17):1897–1916.

Javed Syed Ashar, S. S., and Vineet, G. 2018. Learning unsupervised visual grounding through semantic self-supervision. *arXiv:1803.06506*.

Johnson, J.; Karpathy, A.; and Fei-Fei, L. 2016. Densecap: Fully convolutional localization networks for dense captioning. In *Proc. IEEE CVPR*.

Karpathy, A.; Toderici, G.; Shetty, S.; Leung, T.; Sukthankar, R.; and Fei-Fei, L. 2014. Large-scale video classification with convolutional neural networks. In *Proc. IEEE CVPR*.

Kingma, D. P., and Ba, J. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Krishna, R.; Hata, K.; Ren, F.; Fei-Fei, L.; and Niebles, J. C. 2017. Dense-captioning events in videos. In *Proc. IEEE ICCV*.

Lin, X., and Parikh, D. 2016. Leveraging visual question answering for image-caption ranking. In *Proc. ECCV*.

Lin, T.; Zhao, X.; and Shou, Z. 2017. Single shot temporal action detection. In *Proc. ACM Conference on Multimedia*.

Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.-Y.; and Berg, A. C. 2016. SSD: Single shot multibox detector. In *Proc. ECCV*.

Lu, J.; Yang, J.; Batra, D.; and Parikh, D. 2016. Hierarchical question-image co-attention for visual question answering. In *NIPS*.

Ma, L.; Lu, Z.; Shang, L.; and Li, H. 2015. Multimodal convolutional neural networks for matching image and sentence. In *Proc. IEEE ICCV*.

Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G. S.; and Dean, J. 2013. Distributed representations of words and phrases and their compositionality. In *NIPS*.

Ramanishka, V.; Das, A.; Zhang, J.; and Saenko, K. 2017. Top-down visual saliency guided by captions. In *Proc. IEEE CVPR*.

Ren, S.; He, K.; Girshick, R.; and Sun, J. 2015. Faster R-CNN: Towards real-time object detection with region proposal networks. In *NIPS*.

Rohrbach, A.; Rohrbach, M.; Hu, R.; Darrell, T.; and Schiele, B. 2016. Grounding of textual phrases in images by reconstruction. In *Proc. ECCV*.

Schroff, F.; Kalenichenko, D.; and Philbin, J. 2015. Facenet: A unified embedding for face recognition and clustering. In *Proc. IEEE CVPR*.

Serena Yeung afinally, n. O. R.; Mori, G.; and Fei-Fei, L. 2016. End-to-end Learning of Action Detection from Frame Glimpses in Videos. In *Proc. IEEE CVPR*.

Shih, K. J.; Singh, S.; and Hoiem, D. 2016. Where to look: Focus regions for visual question answering. In *Proc. IEEE CVPR*.

Shou, Z.; Wang, D.; and Chang, S.-F. 2016. Temporal Action Localization in Untrimmed Videos via Multi-stage CNNs. In *Proc. IEEE CVPR*.

Sigurdsson, G. A.; Varol, G.; Wang, X.; Farhadi, A.; Laptev, I.; and Gupta, A. 2016. Hollywood in homes: Crowdsourcing data collection for activity understanding. In *Proc. ECCV*.

Singh, B.; Marks, T. K.; Jones, M.; Tuzel, O.; and Shao, M. 2016. A multi-stream bi-directional recurrent neural network for fine-grained action detection. In *Proc. IEEE CVPR*.

Tran, D.; Bourdev, L.; Fergus, R.; Torresani, L.; and Paluri, M. 2015. Learning spatiotemporal features with 3D convolutional networks. In *Proc. IEEE ICCV*.

Vendrov, I.; Kiros, R.; Fidler, S.; and Urtasun, R. 2016. Order-embeddings of images and language. *ICLR*.

Venugopalan, S.; Rohrbach, M.; Donahue, J.; Mooney, R.; Darrell, T.; and Saenko, K. 2015a. Sequence to sequence – video to text. In *Proc. IEEE ICCV*.

Venugopalan, S.; Xu, H.; Donahue, J.; Rohrbach, M.; Mooney, R.; and Saenko, K. 2015b. Translating videos to natural language using deep recurrent neural networks. In *NAACL-HLT*.

Vinyals, O.; Toshev, A.; Bengio, S.; and Erhan, D. 2015. Show and tell: A neural image caption generator. In *Proc. IEEE CVPR*.

Wang, L.; Li, Y.; Huang, J.; and Lazebnik, S. 2018. Learning two-branch neural networks for image-text matching tasks. *IEEE TPAMI*.

Xiong, C.; Merity, S.; and Socher, R. 2016. Dynamic memory networks for visual and textual question answering. In *Proc. ICML*.

Xu, H., and Saenko, K. 2016. Ask, attend and answer: Exploring question-guided spatial attention for visual question answering. In *Proc. ECCV*.

Xu, H.; Venugopalan, S.; Ramanishka, V.; Rohrbach, M.; and Saenko, K. 2015. A multi-scale multiple instance video description network. *arXiv preprint arXiv:1505.05914*.

Xu, H.; Li, B.; Ramanishka, V.; Sigal, L.; and Saenko, K. 2019. Joint event detection and description in continuous video streams. In *Proc. WACV*.

Xu, H.; Das, A.; and Saenko, K. 2017. R-C3D: Region convolutional 3D network for temporal activity detection. In *Proc. IEEE ICCV*.

Yang, Z.; He, X.; Gao, J.; Deng, L.; and Smola, A. 2016. Stacked attention networks for image question answering. In *Proc. IEEE CVPR*.

Yao, L.; Torabi, A.; Cho, K.; Ballas, N.; Pal, C.; Larochelle, H.; and Courville, A. 2015. Describing videos by exploiting temporal structure. In *Proc. IEEE ICCV*.

Yu, Y.; Ko, H.; Choi, J.; and Kim, G. 2017. End-to-end concept word detection for video captioning, retrieval, and question answering. In *Proc. IEEE CVPR*.