# Localizing Natural Language in Videos

**Jingyuan Chen,**[1*] **Lin Ma,**[2†] **Xinpeng Chen,**[2*] **Zequn Jie,**[2] **Jiebo Luo**[3]
[1]Alibaba Group, [2]Tencent AI Lab, [3]University of Rochester
{jingyuanchen91, forest.linma, jschenxinpeng, zequn.nus}@gmail.com, jluo@cs.rochester.edu

## Abstract

In this paper, we consider the task of natural language video localization (NLVL): given an untrimmed video and a natural language description, the goal is to localize a segment in the video which semantically corresponds to the given natural language description. We propose a localizing network (L-Net), working in an end-to-end fashion, to tackle the NLVL task. We first match the natural sentence and video sequence by cross-gated attended recurrent networks to exploit their fine-grained interactions and generate a sentence-aware video representation. A self interactor is proposed to perform cross-frame matching, which dynamically encodes and aggregates the matching evidences. Finally, a boundary model is proposed to locate the positions of video segments corresponding to the natural sentence description by predicting the starting and ending points of the segment. Extensive experiments conducted on the public TACoS and DiDeMo datasets demonstrate that our proposed model performs effectively and efficiently against the state-of-the-art approaches.

## Introduction

Visual understanding tasks involving language, such as captioning (Chen et al. 2018b; 2018a; Reed et al. 2016; Vinyals et al. 2015; Jiang et al. 2018; Wang et al. 2018b; 2018a), visual question answering (Ma, Lu, and Li 2016; Antol et al. 2015; Xiong, Merity, and Socher 2016; Yang et al. 2016), image and sentence matching (Ma et al. 2015) natural language object retrieval (Hu et al. 2016), have emerged as avenues for expanding the diversity of information that can be recovered from visual contents. With the recent release of the TACoS (Gao et al. 2017) and DiDeMo datasets (Hendricks et al. 2017), the task of natural language video localization (NLVL) has gained considerable attentions. As shown in Fig. 1, the task aims to localize a segment in the video which semantically corresponds to the given natural language description. However, similar to other vision-language tasks, cross-modal interactions and complicated context information issue pose significant challenge to natural language localization in videos.

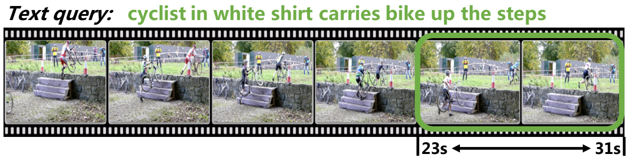

Text query: cyclist in white shirt carries bike up the steps
23s
31s

Figure 1: Natural language video localization is designed to localize a segment (the green box) with a start point (23rd s) and an end point (31st s) in the video given natural language description.

Existing techniques (Gao et al. 2017; Hendricks et al. 2017; Lin et al. 2014; Tellex and Roy 2009) for natural language localization in videos often rely on temporal sliding windows over a video sequence to generate segment candidates, which are then independently compared (Hendricks et al. 2017) or combined (Gao et al. 2017) with the given natural sentence to perform the localization. These models enable a good global matching between the segment candidates and sentences. However, they often suffer from overlooking the fine-grained interactions and limited context information, as well as low efficiency. Specifically, the fine-grained interactions between the frames and words across video-sentence modalities and the rich visual context information are not fully exploited. In addition, these methods are computationally expensive due to the exhaustive search in the temporal domain.

In order to handle the drawbacks, we propose a localization network (L-Net) for the NLVL task. The untrimmed video sequence is processed frame by frame without the need to handle overlapping temporal segments. The key contributions of this work are four-fold:

- We propose a cross-gated attended recurrent network to exploit the fine-grained interactions between the natural sentence and video. In particular, the frame-specific sentence representation is generated by attending the sentence representations with respect to each video frame. Further, a cross gating process is introduced to assign different levels of importance to video (or sentence) parts depending on their relevance to the sentence description (or video content). In this way, the relevant video parts are emphasized while the irrelevant ones are gated out.
- We propose a self interactor to exploit the rich contextual information. We perform cross-frame matching on

[*] Work done while Jingyuan Chen and Xinpeng Chen were Research Interns with Tencent AI Lab.

[†] Corresponding author.

the sentence-aware video representations, which dynamically encodes and aggregates matching evidences from the whole video.

- We propose a novel segment localizer by predicting the starting and ending boundary of the video segment, which semantically corresponds to the given sentence.
- We evaluate our proposed L-Net on TACoS (Gao et al. 2017) and DiDeMo (Hendricks et al. 2017) datasets. Extensive experiments demonstrate the effectiveness and efficiency of our proposed L-Net, which achieves the state-of-the-art performances.

# Related Work

## Temporal Action Detection and Proposals

Temporal action proposals have been proposed to generate temporal window candidates that possibly contain actions. Most previous works perform the proposal generation using a computationally expensive temporal sliding window approach (Duchenne et al. 2009; Oneata, Verbeek, and Schmid 2013) combined with action classifiers trained on multiple features (Tang et al. 2013). Recent works generate spatio-temporal proposals in video, including tubelets (Jain et al. 2014), action tubes (Gkioxari and Malik 2015), and the actionness measure (Chen et al. 2014). To reduce the computational overhead of the sliding window search, some attempts focus on encoding a sequence of visual representations (Buch et al. 2017; Escorcia et al. 2016). Specifically, DAPs (Escorcia et al. 2016) applies Long-Short Term Memory (LSTM) (Hochreiter and Schmidhuber 1997) to encode a video stream into discriminative states, based on which proposals of varied temporal scale are localized via a fixed-length sliding window. However, DAPs still needs to perform computations on overlapping windows. SST (Buch et al. 2017) further reduces the computation by introducing a model that processes each input frame only once and thereby processes the full video in a single pass. However, the temporal action proposal only performs on videos without including the language part, which treats actions as distinct classes, and therefore require a fixed set of action labels. Instead, NLVL solves the task of temporally localizing free-form language in videos with different modalities and more complex context information, which is more flexible and challenging.

## Vision-Language Localization

Cross-modal localization of visual events that match a natural sentence description is a typical vision-language task. The task of natural language object retrieval localizes objects in images given natural sentence description, which is usually formulated as a ranking problem over a set of spatial regions in the image. Different spatial contexts, such as spatial configurations (Hu et al. 2016), attributes (Yu et al. 2018), and relationships between objects (Hu et al. 2017), are incorporated to improve the localization performance. In the video domain some of the representative works (Yu and Siskind 2013; Lin et al. 2014) focus on spatial-temporal language localization. The semantics of sentences is matched to visual concepts via exploiting object appearance, motion and spatial relationships. However, these are limited to a small set of nouns. To learn the semantics of natural language, the late fusion is performed at the sentence level: the natural language is embedded into a single vector and then combined with the video feature vector. Therefore, the important temporal information about word sequences is lost.

Recently, larger datasets (Gao et al. 2017; Hendricks et al. 2017) are built to support more flexible localizations. These methods measure the similarity between video segment and natural language via a common embedding space. The existing localization mechanisms are either inefficient (sliding-window based) or inflexible (hard-coded) (Xu et al. 2018). First, the video segment generation process is computationally expensive, as they carry out overlapping sliding window matching (Gao et al. 2017) or exhaustive search (Hendricks et al. 2017). Second, the evolving fine-grained video-sentence interactions between words and video frames are ignored, where simple concatenation (Gao et al. 2017) or squared distance loss (Hendricks et al. 2017) is used. In contrast with these approaches, we propose a single stream framework L-Net which takes advantages of the fine-grained interactions between two modalities and the evidences from the context to semantically localize the video segment given the natural language.

# Methods

Given a video $V$ and a natural language query $S$, the NLVL task aims at identifying a video segment with the starting position $\tau^s$ and ending position $\tau^e$ as the localization, which corresponds to the natural language sentence. The framework of our proposed L-Net for tackling the NLVL task is illustrated in Fig. 2, which consists of the following four components:

- The **encoder** utilizes bi-directional recurrent neural networks (RNNs), specifically the gated recurrent units (GRUs) (Rohrbach et al. 2016) specializing in processing long-term dependencies of sequential data, to encode the sentence and video sequence, respectively.
- The **cross modal interactor** attentively fuses the sentence and video and comprehensively exploits their relationships in a fine-grained manner.
- The **self interactor** performs cross-frame matching on the generated sentence-aware video representations to dynamically encode and aggregate the matching evidences over the whole video.
- The **segment localizer** predicts the starting and ending boundary of the video segment, which semantically corresponds to the given sentence.

## Video and Sentence Encoder

We first utilize one image CNN to encode each video frame into a feature representation. With the encoded frame features $V = \{f_t\}_{t=1}^{T}$ and word embeddings of the sentence $S = \{w_n\}_{n=1}^{N}$, two bi-directional RNNs are used to sequentially process the two different modalities and produce new representations for all video frames and all words in sentences, respectively. Specifically, we use GRU, which per-

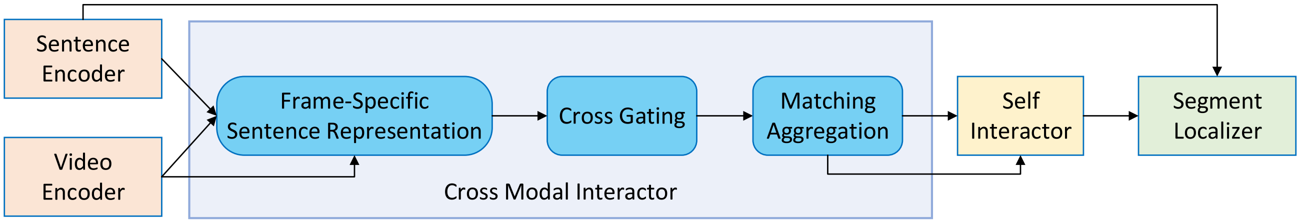


Figure 2: The architecture of the proposed L-Net, which consists of four components, namely the encoder, cross modal interactor, self interactor, and segment localizer. For the cross modal interactor, frame-specific sentence representation is generated by attending the sentence representations with respect to each video frame. The cross grating mechanism is performed to enhance the fine-grained matching behaviors between video and sentence, which is further aggregated temporally by the matching aggregation module.

forms similarly to long short-term memory (LSTM) (Hochreiter and Schmidhuber 1997) but is computationally cheaper.

$$\begin{aligned} \mathbf{H}^V &= \text{B-GRU}_v(V), \\ \mathbf{H}^S &= \text{B-GRU}_s(S). \end{aligned} \tag{1}$$

According to the characteristics of bi-directional GRU (B-GRU), the $i$-th column vector $\mathbf{h}_i^v$ (or $\mathbf{h}_i^s$) in $\mathbf{H}^V$ (or $\mathbf{H}^S$) represents the $i$-th frame (or word) in the video (or the sentence) with consideration of the contextual information from both forward and backward directions.

## Cross Modal Interactor

Based on the obtained representations from the video and sentence encoders, we design a cross modal interactor to capture the fine-grained interactions between the video frames and words, which characterizes the matching behaviors across sentence and video.

**Frame-Specific Sentence Representation.** In order to exploit the fine-grained interactions between video and sentence, we introduce a series of attentively weighted combinations of the hidden states of sentence, where each combination is specifically generated for a particular video frame. We use $\bar{\mathbf{h}}_t^s$ to denote such an attentive representation for sentence $S$ at time step $t$ with respect to the $t$-th video frame, which is defined as follows:

$$\bar{\mathbf{h}}_t^s = \textstyle\sum_{n=1}^N \alpha_t^n \mathbf{h}_n^s, \tag{2}$$

where $\alpha_t^n$ is an attention weight that encodes the degree to which the $n$-th word in the sentence is matched with the $t$-th video frame. The widely used soft-attention mechanism (Chen et al. 2017) is adopted to generate the attention weights:

$$\begin{aligned} a_t^n &= \mathbf{w}_r^\intercal \tanh(\mathbf{W}_r^S \mathbf{h}_n^s + \mathbf{W}_r^V \mathbf{h}_t^v), \\ \alpha_t^n &= \frac{\exp(a_t^n)}{\sum_{j=1}^N \exp(a_t^j)}, \end{aligned} \tag{3}$$

where the vector $\mathbf{w}_r$ and matrices $\mathbf{W}_r^*$ are the parameters to be learned. It can be observed that the attention weight $a_t^n$ with respect to the current video frame $\mathbf{h}_t^v$ dynamically changes as the video proceeds. Therefore, such a frame-specific sentence representation receives varying attentive information from all words, guided by the changing frames in a video. As such, the frame-specific sentence representations summarize the relationships between all the video frames and all the words in the sentence.

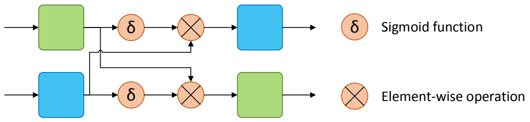


Figure 3: The cross gating module.

**Cross Gating.** Based on the frame-specific sentence representation $\{\bar{\mathbf{h}}_t^s\}_{t=1}^T$ and frame representation $\{\mathbf{h}_t^v\}_{t=1}^T$, we propose a cross gating scheme, as shown in Fig. 3, to gate out the irrelevant parts and emphasize the relevant and informative parts:

$$\begin{aligned} \mathbf{g}_t^v &= \sigma(\mathbf{W}_g^V \mathbf{h}_t^v), \\ \widetilde{\mathbf{h}}_t^s &= \bar{\mathbf{h}}_t^s \odot \mathbf{g}_t^v, \\ \mathbf{g}_t^s &= \sigma(\mathbf{W}_g^S \bar{\mathbf{h}}_t^s), \\ \widetilde{\mathbf{h}}_t^v &= \mathbf{h}_t^v \odot \mathbf{g}_t^s, \end{aligned} \tag{4}$$

where $\mathbf{W}_g^*$ represent the learnable parameters and $\sigma$ denotes the non-linear sigmoid function. It can be observed that the cross gating mechanism controls the extent to which one modality interacts with the other one. Specifically, if the video feature $\mathbf{h}_t^v$ is irrelevant to the query sentence $\bar{\mathbf{h}}_t^s$, both the video feature and sentence representation are filtered to reduce their effect on the subsequent processes. If the two are closely related, the cross gating strategy is expected to further enhance their interactions.

**Matching Aggregation.** With the frame-specific sentence representations and cross gating, the fine-grained matching relationships between video frame and word in sentence are comprehensively exploited. We concatenate the $t$-th video hidden state $\widetilde{\mathbf{h}}_t^v$ and the $t$-th frame-specific sentence feature $\widetilde{\mathbf{h}}_t^s$ as: $\mathbf{b}_t = [\widetilde{\mathbf{h}}_t^v, \widetilde{\mathbf{h}}_t^s]$. Then a bidirectional GRU working on

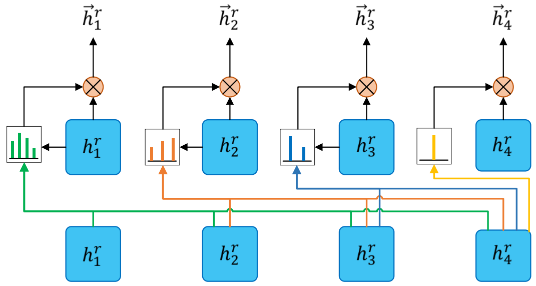


Figure 4: The process of forward attention generation in the self interactor.

$\mathbf{b}_t$ is utilized to further temporally aggregate the matching behaviors between the video frames and words in sentence:

$$\mathbf{h}_t^r = \text{GRU}(\mathbf{b}_t), \tag{5}$$

where $\mathbf{h}_t^r$ is the yielded hidden state, which can be viewed as a sentence-aware video representation, encoding the fine-grained interactions between the two modalities. Due to the inherent properties and characteristics of RNN, important cues regarding localization will be "remembered", while non-essential ones will be "forgotten".

## Self Interactor

In addition to the fine-grained interactions between the video and sentence, the visual context information from other frames also plays an important role to accurately localize the video segment corresponding to the sentence query. Taking the sentence query of "`the first girl in pink walks by the camera`" as an example, the term "`first`" requires temporal context outside its surrounding window for proper inference. Although the sentence-aware video representation $\{\mathbf{h}_t^r\}_{t=1}^T$ generated from cross-modal interactor contains important clues for the NLVL task, one weakness is that the context is not fully considered.

Furthermore, the information accumulated from different directions plays different roles when predicting the starting and ending points of the boundary. Suppose, we predict the probability of a specific frame of being the starting point. Naturally, the visual information after the frame is desired to be accumulated to see if a complete action instance just starts at this frame, and vice versa for predicting the ending point.

Considering the aforementioned issues, we propose a boundary-aware self interactor which performs a cross-frame matching on the sentence-aware video representation. For predicting the starting point, the self interactor first dynamically collects the matching evidences from frames after time step $t$ as:

$$\overrightarrow{\mathbf{h}}_t^r = \textstyle\sum_{i=t}^T \overrightarrow{\beta}_t^i \mathbf{h}_i^r, \tag{6}$$

where $\overrightarrow{\beta}_t^i$ is the attention weight obtained via soft-attention over the set of frames which come after the $t$-th frame, as shown in Fig. 4. We name the $\overrightarrow{\beta}_t^i$ as the forward attention in the following discussion, which is defined as:

$$\begin{aligned} b_t^i &= \mathbf{w}_u^\intercal \tanh(\mathbf{W}_u^V \mathbf{h}_i^r + \mathbf{W}_u^{\widetilde{V}} \mathbf{h}_t^r), \\ \overrightarrow{\beta}_t^i &= \frac{\exp(b_t^i)}{\sum_{j=t}^T \exp(b_t^j)}. \end{aligned} \tag{7}$$

Afterwards, the self interactor aggregates the forward context evidences together:

$$\overrightarrow{\mathbf{h}}_t^d = \text{GRU}([\mathbf{h}_t^r, \overrightarrow{\mathbf{h}}_t^r], \overrightarrow{\mathbf{h}}_{t-1}^d), \tag{8}$$

where the input of GRU is obtained by concatenating the sentence-aware video representation and the obtained context evidences. $\overrightarrow{\mathbf{h}}_t^d$ denotes the yielded forward context-aware video representation. When predicting the ending point, the backward attention weight $\overleftarrow{\beta}_t^i$, the backward accumulated matching evidence $\overleftarrow{\mathbf{h}}_t^r = \sum_{i=1}^t \overleftarrow{\beta}_t^i \mathbf{h}_i^r$, and the backward context-aware video representation $\overleftarrow{\mathbf{h}}_t^d$ are generated in the same way. Next, the segment localizer takes the context-aware video representations as input to perform the localization in the video sequence.

## Segment Localizer

We propose a boundary model which predicts the starting and ending time steps, with the video segment lying between considered to be the localization. We first utilize the attentive sentence vector $\mathbf{h}_o^s = \sum_{i=1}^N c_i \mathbf{h}_i^s$ as the initial state of the segment localizer, where $c_i$ is the attention weight obtained by a self attention strategy:

$$c_i = \frac{\exp(\mathbf{w}_q^\intercal \tanh(\mathbf{W}_q^H \mathbf{h}_i^s + \mathbf{u}))}{\sum_{n=1}^N \exp(\mathbf{w}_q^\intercal \tanh(\mathbf{W}_q^H \mathbf{h}_n^s + \mathbf{u}))}. \tag{9}$$

Given the context-aware video representation $\{\overrightarrow{\mathbf{h}}_t^d\}_{t=1}^T$ and $\{\overleftarrow{\mathbf{h}}_t^d\}_{t=1}^T$ generated from the self interactor of both directions, the attention mechanism is utilized as a pointer to select the starting position $\tau^s$ and ending position $\tau^e$ from the video, respectively:

$$\begin{aligned} s_t^1 &= \frac{\exp(\mathbf{w}_p^\intercal \tanh(\mathbf{W}_p^H \overrightarrow{\mathbf{h}}_t^d + \mathbf{W}_p^{\widetilde{H}} \mathbf{h}_o^s))}{\sum_{i=1}^T \exp(\mathbf{w}_p^\intercal \tanh(\mathbf{W}_p^H \overrightarrow{\mathbf{h}}_i^d + \mathbf{W}_p^{\widetilde{H}} \mathbf{h}_o^s))}, \\ s_t^2 &= \frac{\exp(\mathbf{w}_p^\intercal \tanh(\mathbf{W}_p^H \overleftarrow{\mathbf{h}}_t^d + \mathbf{W}_p^{\widetilde{H}} \mathbf{h}_o^s))}{\sum_{i=1}^T \exp(\mathbf{w}_p^\intercal \tanh(\mathbf{W}_p^H \overleftarrow{\mathbf{h}}_i^d + \mathbf{W}_p^{\widetilde{H}} \mathbf{h}_o^s))}, \\ \tau^s &= \arg\max(s_1^1, \ldots, s_T^1), \\ \tau^e &= \arg\max(s_1^2, \ldots, s_T^2). \end{aligned} \tag{10}$$

## Training

As illustrated in Fig. 2, all the components of our proposed L-Net, namely the sentence/video encoders, cross modal interactor, self interactor, and segment localizer, couple together and can be trained in an end-to-end fashion. In this paper, we train our proposed L-Net by minimizing the sum of the negative log probabilities (multiclass cross-entropy) of the ground truth starting and ending positions by the predicted distributions.

### Testing.

During the testing phase, each segment candidate with the starting position $t_1$ and ending position $t_2$ will be assigned a score $s = s^1_{t_1} \times s^2_{t_2}$, which indicates the probability that the video segment corresponds to given sentence $S$. Finally, the evaluation is reduced to a ranking problem over all the video segment candidates based on the generated scores.

## Experiments

We evaluate the proposed L-Net on two public video localization datasets (TACoS (Gao et al. 2017) and DiDeMo (Hendricks et al. 2017)), which contain videos as well as their associated temporally annotated sentences. We describe the dataset, evaluation metrics, and implementation details before we present the quantitative results, the ablation study, and the qualitative results.

### Dataset

**TACoS**[1]. It has $127$ videos with an average length of $5.84$ minutes, selected from the MPII Cooking Composite Activities video corpus (Rohrbach et al. 2012). We follow the same split as in  (Gao et al. 2017), which has 10146, 4589, and $4083$ video-sentence pairs for training, validation, and testing respectively.

**DiDeMo**[2]. It has $10464$ 25-50 second long videos, selected from YFCC100M (Thomee et al. 2015). We use the same split provided by  (Hendricks et al. 2017) for a fair comparison, which has $33008$, $4180$, and $4022$ video-sentence pairs for training, validation, and testing respectively.

The two datasets serve as a good testbed as they contain challenging variations, such as complex query and videos of various lengths.

### Evaluation Metrics

**Intersection over union**. We use the mean intersection over union (mIoU) metric which calculates the average IoU among all testing samples. The IoU metric is particularly challenging for short video groundings.

**Recall**. We adopt "R@$n$, IoU=$m$" proposed by  (Hu et al. 2016) as the other evaluation metric, which represents the percentage of testing samples which have at least one of the top-$n$ results with IoU larger than $m$.

### Implementation Details

The video feature is usually generated with a time resolution. We sample every 5 second as done by (Hendricks et al. 2017). In particular, since the videos in DiDeMo are only 25-30 second long, the video length is reduced to 6 chunks after sampling. In total there are only $\mathrm{C}_7^2 = 7 \times 6/2 = 21$ different ways of localization for DiDeMo videos. To be consistent with the baseline methods, the experiments on the DiDeMo dataset are conducted based on optical flow features (Wang et al. 2016) and the experiments on TACoS are based on C3D features (Tran et al. 2015).

For word-level representations, we tokenize each sentence by Stanford CoreNLP (Manning et al. 2014) and use the $300$-D word embeddings from GloVe (Pennington, Socher, and Manning 2014) to initialize the models. The words not found in GloVe are initialized as zero vectors. Please note that the word embeddings are not fine-tuned during the training phase.

The hidden state dimension $D$ of all layers (including the video, sentence, and interaction GRUs) are set to 75. The mini-batch size is set to 32 for TACoS and 64 for DiDeMo. We use the Adam (Kingma and Ba 2014) optimizer with $\beta_1$ = 0.5 and $\beta_2$ = 0.999. The initial learning rate is set to $0.001$. We train the network for 200 iterations, and the learning rate is gradually decayed over time. We use bi-directional GRU of 3 layers to encode videos and sentences. Dropout (Srivastava et al. 2014) of rate $0.3$ and $0.5$ are utilized.

### Quantitative Evaluation

We compare the performance of our approach with several state-of-the-art benchmarks, specifically CTRL (Gao et al. 2017), MCN (Hendricks et al. 2017), VSA-RNN (Karpathy and Li 2015), and VSA-STV (Karpathy and Li 2015). CTRL generates fused representations via element-wise operations among video segment and sentence representations, and utilizes a temporal regression network to produce the alignment scores and location offsets. MCN learns a shared embedding for video clip-level features and language features. The video features integrate local and global features. We do not compare with the temporal endpoint features as in (Hendricks et al. 2017), since these directly correspond to dataset priors and do not reflect a model's temporal reasoning capability (Liu et al. 2018). VSA-RNN is a sentence based video retrieval method where both the video segment and sentence are encoded by pre-trained models with cosine distance evaluating their similarity. VSA-STV is similar with VSA-RNN. Instead of using RNN to extract the sentence description embedding, VSA-STV uses an off-the-shelf Skip-thoughts (Kiros et al. 2015) sentence embedding extractor. Fig. 5 shows the performance of R@1 and R@5 with respect to the IoU ranges from $0.1$ to $0.9$. Due to the low efficiency of the enumeration-based method MCN, the performance of MCN on the long video dataset TACoS is omitted in Fig. 5.

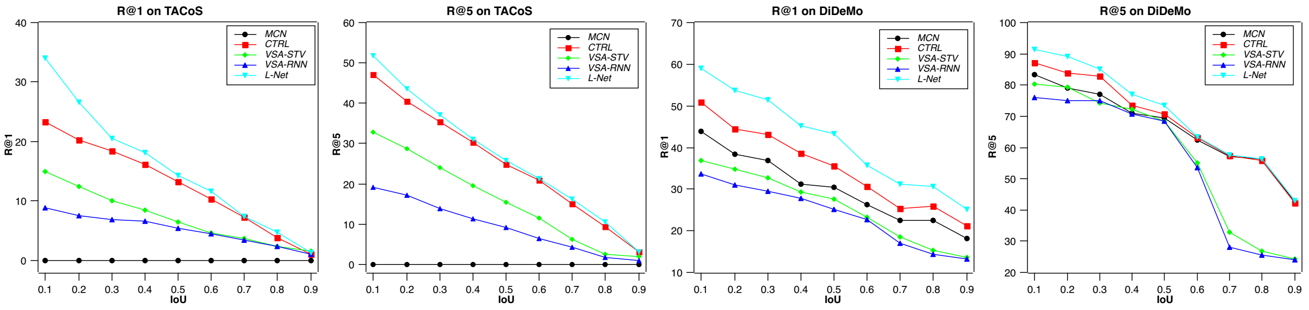
L-Net achieves the best performance on the long video dataset TACoS as well as the short video dataset DiDeMo with respect to R@1 and R@5, which verifies the effectiveness of the proposed framework.

VSA-STV and VSA-RNN achieve poor performance since they ignore both the cross-modal interaction and the context information. They model the isolated video segments with LSTM hence fail to exploit the temporal cues. Moreover, the simple cosine similarity model cannot well capture the interactions between two modalities.

MCN is designed as an enumeration-based approach. In particular, MCN predicts the localization by ranking the $\mathrm{C}_7^2 = 7 \times 6/2 = 21$ limited (*i.e.*, 21) segments in each DiDeMo video. Therefore, although MCN can be effectively applied to videos with several chunks (*e.g.*, DiDeMo), it is not practical for untrimmed long videos (*e.g.*, TACoS). MCN incorporates the context information by utilizing the average pooling of

[1] https://github.com/jiyanggao/TALL.

[2] https://github.com/LisaAnne/LocalizingMoments.



(a) R@1 on TACoS (b) R@5 on TACoS (c) R@1 on DiDeMo (d) R@5 on DiDeMo

Figure 5: Performance of R@$n$, IoU=$m$ where $n$ values at 1 and 5 and $m$ ranges from 0.1 to 0.9 with interval of 0.1 on the TACoS and DiDeMo datasets.

Table 1: Contributions of different components of our algorithm evaluated on the TACoS and DiDeMo datasets in terms of mIoU(%).The enabled components are marked by ✓.

| Component | | Enable/Disable | | | | |
|---|---|---|---|---|---|---|
| Cross Modal Interactor | FS | | ✓ | ✓ | ✓ | ✓ |
| | CG | ✓ | | ✓ | ✓ | ✓ |
| Self Interactor | SI | ✓ | ✓ | | ✓ | ✓ |
| | UA | | | | ✓ | |
| | FB | ✓ | ✓ | | | ✓ |
| TACoS | | 11.97 | 12.43 | 12.56 | 12.98 | **13.41** |
| DiDeMo | | 38.95 | 40.16 | 38.74 | 41.02 | **41.43** |

Table 2: Efficiency comparison with respect to FPS.

| | CTRL | MCN | L-NET |
|---|---|---|---|
| FPS | 562 | 286 | 1,032 |

the context segment frame features, ignoring the adaptive importance of the context. This is the reason why MCN achieves worse performance compared with L-Net and CTRL on DiDeMo.

CTRL performs better on the DiDeMo dataset compared with MCN. The reason is that CTRL is capable of exploiting the interactions across the visual and textual modalities through element-wise operation.

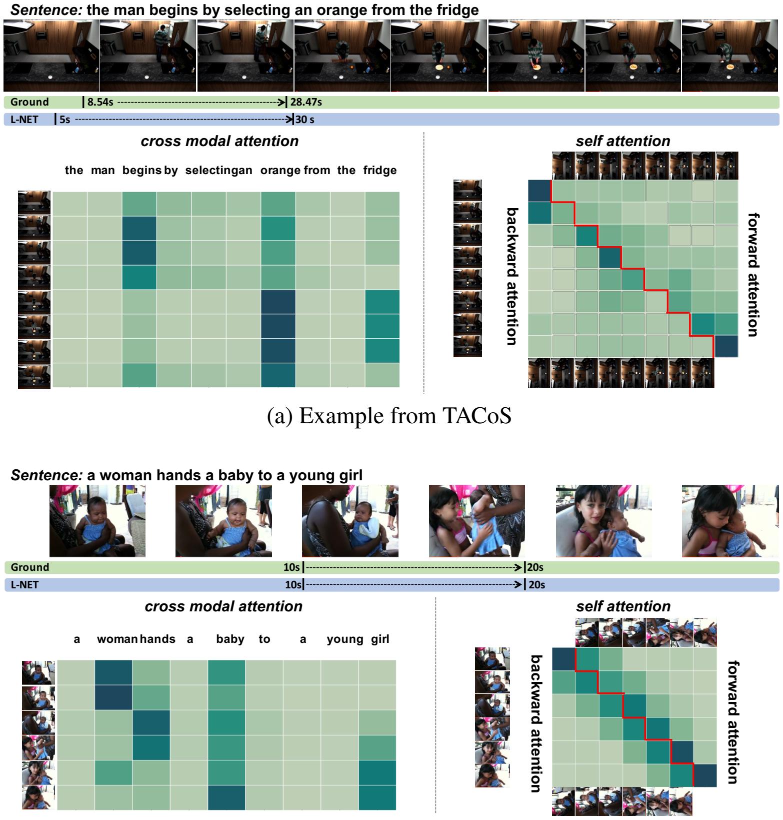
**Efficiency.** We also evaluate the efficiency of our proposed L-Net, by comparing its runtime against CTRL and MCN. Table 2 shows the frames per second (FPS) for different methods, which excludes the feature extraction time and evaluation time. Compared with CTRL and MCN, our L-Net model significantly reduce the localization time. The reason is that the proposed L-Net processes each video as one single stream without evaluating on overlapping sliding windows, while CTRL and MCN methods adopt the typical scan and localize architecture, often need to sample densely overlapped video segment candidates by various sliding windows. All the experiments are conducted on a Tesla M40 GPU.

## Ablation Study

We validate the contributions of the components in our method by presenting an ablation study summarized in Table 1 on the two datasets. We mark the enabled components using the "✓" symbol. We analyze the contribution of the cross modal interaction, including the frame-specific sentence feature ('FS') and the cross gating mechanism ('CG'). In addition, we analyze the effects of the dynamic self interactor ('SI'). Specifically, we assess the performance changes of two configurations in self interactor: (i) incorporating visual context among all the video frames without considering attention in different directions ('UA') and (ii) utilizing the combination of forward and backward attention ('FB') as described in Section `Self Interactor`.

As illustrated in Table 1, we generally observe that both the cross-modal interaction between two modalities and the self attention within the whole video are important for the NLVL task as they dynamically enrich the video representation and aggregate the matching behaviors from both modalities.

For the cross-modal interactor, removing frame-specific sentence (Disable both 'FS' and 'UA') results in large mIoU drop, which reveals that it is necessary to discriminate the contribution of each word in a sentence query when performing localization. It can be observed that when disabling the cross gating (Disable both 'CG' and 'UA'), the prediction performance decreased, which demonstrates that the cross gating contributes towards the model's performance. In particular, cross gating can help filter out the irrelevant information meanwhile enhancing the meaningful interactions between



(a) Example from TACoS

(b) Example from DiDeMo

Figure 6: Some examples of our L-Net on the NLVL task with the corresponding heatmaps of the attention weights. The darker the color is, the larger its represented attention weight is.

the sentence and videos, which can thereby benefit the final localization.

For the self interactor, we first show that the performance of the model drops when disabling the self interactor (Disable 'SI', 'UA', and 'FB'). This is due to the fact that the contextual information in the video plays an important role. For the attention mechanism adopted within self interactor, the bi-directional attention (Disable 'UA') performs better than using the non-directional attention (Disable 'FB'). This result indicates that when predicting the boundaries in the localization, the directional context information plays an important role.

### Qualitative evaluation

Finally, we show some examples to visualize the localization results in Fig. 6 as well as the corresponding heatmaps of cross-modal and self attention weights. The cross-modal attention is at word-by-frame level. It can be observed that some words well match the frames. For example in Fig. 6 (a), the temporal indicator "`begins`" obtains higher attention among the first 3 frames which is consistent with the prediction result. Although the word "`orange`" appears across all the frames, the 5-th to 8-th frames obtain higher attention than the first 4 frames. The self attention is at frame-by-frame level, where each frame attentively matches other frames. It can be observed that our self attention focus on the related frames in the neighborhood.

## Conclusion

We present an end-to-end localization network (L-Net) for the task of natural language localization in videos. With the proposed cross modal interactor and the self interactor, our approach takes advantages of the fine-grained interactions between two modalities and the evidences from the context to semantically localize the video segment corresponding to the natural sentence. Extensive experiments on two real-world datasets demonstrate the effectiveness and efficiency of the proposed L-Net.

## References

Antol, S.; Agrawal, A.; Lu, J.; Mitchell, M.; Batra, D.; Zitnick, C. L.; and Parikh, D. 2015. VQA: visual question answering. In *ICCV*, 2425–2433.

Buch, S.; Escorcia, V.; Shen, C.; Ghanem, B.; and Niebles, J. C. 2017. Sst: Single-stream temporal action proposals. In *CVPR*.

Chen, W.; Xiong, C.; Xu, R.; and Corso, J. J. 2014. Actionness ranking with lattice conditional ordinal random fields. In *CVPR*.

Chen, J.; Zhang, H.; He, X.; Nie, L.; Liu, W.; and Chua, T. 2017. Attentive collaborative filtering: Multimedia recommendation with item- and component-level attention. In *SIGIR*, 335–344.

Chen, J.; Chen, X.; Ma, L.; Jie, Z.; and Chua, T.-S. 2018a. Temporally grounding natural sentence in video. In *EMNLP*.

Chen, X.; Ma, L.; Jiang, W.; Yao, J.; and Liu, W. 2018b. Regularizing rnns for caption generation by reconstructing the past with the present. *CVPR*.

Duchenne, O.; Laptev, I.; Sivic, J.; Bach, F. R.; and Ponce, J. 2009. Automatic annotation of human actions in video. In *ICCV*.

Escorcia, V.; Heilbron, F. C.; Niebles, J. C.; and Ghanem, B. 2016. DAPs: Deep action proposals for action understanding. In *ECCV*.

Gao, J.; Sun, C.; Yang, Z.; and Nevatia, R. 2017. TALL: temporal activity localization via language query. In *ICCV*, 5277–5285.

Gkioxari, G., and Malik, J. 2015. Finding action tubes. In *CVPR*.

Hendricks, L. A.; Wang, O.; Shechtman, E.; Sivic, J.; Darrell, T.; and Russell, B. 2017. Localizing moments in video with natural language. In *ICCV*, 5804–5813.

Hochreiter, S., and Schmidhuber, J. 1997. Long short-term memory. *Neural Computation* 9(8):1735–1780.

Hu, R.; Xu, H.; Rohrbach, M.; Feng, J.; Saenko, K.; and Darrell, T. 2016. Natural language object retrieval. In *CVPR*, 4555–4564.

Hu, R.; Rohrbach, M.; Andreas, J.; Darrell, T.; and Saenko, K. 2017. Modeling relationships in referential expressions with compositional modular networks. In *CVPR*.

Jain, M.; van Gemert, J. C.; Jégou, H.; Bouthemy, P.; and Snoek, C. G. M. 2014. Action localization with tubelets from motion. In *CVPR*.

Jiang, W.; Ma, L.; Jiang, Y.-G.; Liu, W.; and Zhang, T. 2018. Recurrent fusion network for image captioning. In *ECCV*.

Karpathy, A., and Li, F. 2015. Deep visual-semantic alignments for generating image descriptions. In *CVPR*.

Kingma, D. P., and Ba, J. 2014. Adam: A method for stochastic optimization. *CoRR* abs/1412.6980.

Kiros, R.; Zhu, Y.; Salakhutdinov, R.; Zemel, R. S.; Urtasun, R.; Torralba, A.; and Fidler, S. 2015. Skip-thought vectors. In *NIPS*, 3294–3302.

Lin, D.; Fidler, S.; Kong, C.; and Urtasun, R. 2014. Visual semantic search: Retrieving videos via complex textual queries. In *CVPR*, 2657–2664.

Liu, B.; Yeung, S.; Chou, E.; Huang, D.-A.; Fei-Fei, L.; and Carlos Niebles, J. 2018. Temporal modular networks for retrieving complex compositional activities in video. In *ECCV*, 552–568.

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

Ma, L.; Lu, Z.; and Li, H. 2016. Learning to answer questions from image using convolutional neural network. In *AAAI*.

Manning, C. D.; Surdeanu, M.; Bauer, J.; Finkel, J. R.; Bethard, S.; and McClosky, D. 2014. The stanford corenlp natural language processing toolkit. In *ACL*, 55–60.

Oneata, D.; Verbeek, J. J.; and Schmid, C. 2013. Action and event recognition with fisher vectors on a compact feature set. In *ICCV*.

Pennington, J.; Socher, R.; and Manning, C. D. 2014. Glove: Global vectors for word representation. In *EMNLP*, 1532–1543.

Reed, S. E.; Akata, Z.; Yan, X.; Logeswaran, L.; Schiele, B.; and Lee, H. 2016. Generative adversarial text to image synthesis. In *ICML*, 1060–1069.

Rohrbach, M.; Regneri, M.; Andriluka, M.; Amin, S.; Pinkal, M.; and Schiele, B. 2012. Script data for attribute-based recognition of composite activities. In *ECCV*.

Rohrbach, A.; Rohrbach, M.; Hu, R.; Darrell, T.; and Schiele, B. 2016. Grounding of textual phrases in images by reconstruction. In *ECCV*, 817–834. ACL.

Srivastava, N.; Hinton, G. E.; Krizhevsky, A.; Sutskever, I.; and Salakhutdinov, R. 2014. Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research* 15(1):1929–1958.

Tang, K. D.; Yao, B.; Li, F.; and Koller, D. 2013. Combining the right features for complex event recognition. In *ICCV*.

Tellex, S., and Roy, D. 2009. Towards surveillance video search by natural language query. In *CIVR*.

Thomee, B.; Shamma, D. A.; Friedland, G.; Elizalde, B.; Ni, K.; Poland, D.; Borth, D.; and Li, L. 2015. The new data and new challenges in multimedia research. *CoRR* abs/1503.01817.

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

Vinyals, O.; Toshev, A.; Bengio, S.; and Erhan, D. 2015. Show and tell: A neural image caption generator. In *CVPR*, 3156–3164.

Wang, L.; Xiong, Y.; Wang, Z.; Qiao, Y.; Lin, D.; Tang, X.; and Gool, L. V. 2016. Temporal segment networks: Towards good practices for deep action recognition. In *ECCV*.

Wang, B.; Ma, L.; Zhang, W.; and Liu, W. 2018a. Reconstruction network for video captioning. In *CVPR*.

Wang, J.; Jiang, W.; Ma, L.; Liu, W.; and Xu, Y. 2018b. Bidirectional attentive fusion with context gating for dense video captioning. In *CVPR*.

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

Xu, H.; He, K.; Sigal, L.; Sclaroff, S.; and Saenko, K. 2018. Text-to-clip video retrieval with early fusion and re-captioning. *CoRR* abs/1804.05113.

Yang, Z.; He, X.; Gao, J.; Deng, L.; and Smola, A. J. 2016. Stacked attention networks for image question answering. In *CVPR*, 21–29.

Yu, H., and Siskind, J. M. 2013. Grounded language learning from video described with sentences. In *ACL*.

Yu, L.; Lin, Z.; Shen, X.; Yang, J.; Lu, X.; Bansal, M.; and Berg, T. L. 2018. Mattnet: Modular attention network for referring expression comprehension. *CoRR* abs/1801.08186.