Unsupervised Temporal Video Grounding with Deep Semantic Clustering

Daizong Liu[1,2†], Xiaoye Qu[2†], Yinzhen Wang[3†], Xing Di[4], Kai Zou[4], Yu Cheng[5], Zichuan Xu[6], Pan Zhou[1*]

[1]The Hubei Engineering Research Center on Big Data Security, School of Cyber Science and Engineering,
Huazhong University of Science and Technology
[2]School of Electronic Information and Communication, Huazhong University of Science and Technology
[3]School of Computer Science and Technology, Huazhong University of Science and Technology
[4]ProtagoLabs Inc [5]Microsoft Research [6]Dalian University of Technology
{dzliu, xiaoye, yinzhenwang, panzhou}@hust.edu.cn, {xing.di, kz}@protagolabs.com,
yu.cheng@microsoft.com, z.xu@dlut.edu.cn

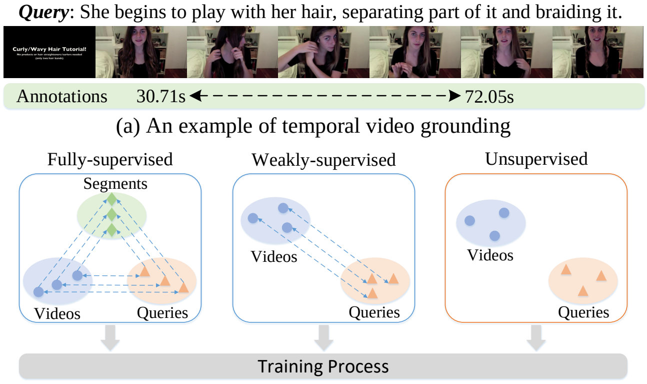
Abstract

Temporal video grounding (TVG) aims to localize a target segment in a video according to a given sentence query. Though respectable works have made decent achievements in this task, they severely rely on abundant video-query paired data, which is expensive and time-consuming to collect in real-world scenarios. In this paper, we explore whether a video grounding model can be learned without any paired annotations. To the best of our knowledge, this paper is the first work trying to address TVG in an unsupervised setting. Considering there is no paired supervision, we propose a novel **D**eep **S**emantic **C**lustering **N**etwork (DSCNet) to leverage all semantic information from the whole query set to compose the possible activity in each video for grounding. Specifically, we first develop a language semantic mining module, which extracts implicit semantic features from the whole query set. Then, these language semantic features serve as the guidance to compose the activity in video via a video-based semantic aggregation module. Finally, we utilize a foreground attention branch to filter out the redundant background activities and refine the grounding results. To validate the effectiveness of our DSCNet, we conduct experiments on both ActivityNet Captions and Charades-STA datasets. The results demonstrate that DSCNet achieves competitive performance, and even outperforms most weakly-supervised approaches.

Introduction

Temporal video grounding (TVG) is an important yet challenging task in video understanding, which has drawn increasing attention due to its vast potential applications, such as activity detection (Zhao et al. 2017) and human-computer interaction (Singha, Roy, and Laskar 2018). As depicted in Figure 1 (a), it aims to localize a segment in a video according to the semantic of a sentence query.

Most previous methods (Wang, Ma, and Jiang 2020a; Liu et al. 2021b; Zhang et al. 2019; Liu et al. 2020b; Wang, Ma, and Jiang 2020b; Chen et al. 2020) proposed for this task are under fully-supervised setting. Some of them (Liu et al. 2021b; Zhang et al. 2020a; Liu et al. 2020b) match the pre-defined segment proposals with the query and then select the best candidate. Others (Chen et al. 2019; Mun, Cho, and Han 2020; Zeng et al. 2020; Chen et al. 2020) directly predict the temporal boundary of the video segment. However, these methods are data-hungry, requiring a large amount of fully annotated data. For instance, the widely used ActivityNet Captions dataset contains 20,000 videos and 100,000 matched sentence queries with their corresponding segment boundaries. Manually annotating such a huge amount of data is very time-consuming and labor-intensive. To alleviate this problem, recent works explore a weakly-supervised setting (Bojanowski et al. 2015; Mithun, Paul, and Roy-Chowdhury 2019; Song et al. 2020; Zhang et al. 2020b) where only paired videos and queries are available in the training stage. Though they leave out the segment annotations in training, these methods still need to access the abundant knowledge of the matched video-query pairs.



(a) An example of temporal video grounding

(b) Different settings in temporal video grounding

Figure 1: (a) An example of temporal video grounding. (b) Different from the fully-supervised (paired video-query and detailed segment annotations) and weakly-supervised (only paired video-query knowledge) settings in TVG, there is no supervised information in unsupervised setting.

In this paper, we focus on how to learn a video grounding model without any supervision, which excludes both paired video-query knowledge and corresponding segment annotations, as shown in Figure 1 (b). Considering there is no annotated information, what we can access to is only the internal information in the queries and videos. As different words or

[†]Equal contributions. *Corresponding author.

phrases in different queries may share potentially similar semantic, we can mine all deep semantic representations from the whole query set, and then compose the possible activities in each video according to these language semantic for further grounding. Therefore, the crucial and challenging part of our work lies in how to capture the deep semantic features of the queries and how to aggregate different semantics for composing the contents of the target segments.

To this end, we propose a novel approach to solve this problem, called **D**eep **S**emantic **C**lustering **N**etwork (DSCNet), which mines the deep semantic features from the query set to compose possible activities in each video. Specifically, we first leverage an encoder-decoder model to build a language-based semantic mining module for query encoding, where the learned hidden states are taken as the extracted deep semantic features. In particular, we collect such semantic features from the whole query set and then cluster them to different semantic centers, where features of similar meanings are adjacent. Subsequently, a video-based semantic aggregation module, containing a specific attention branch and a foreground attention branch, is further developed to compose corresponding activities guided by the extracted deep semantic features. For the specific attention branch, it aggregates different semantic for matching and composing the contents of the activity segments. We utilize this branch to generate better video representations by ensuring that the composed activities containing the same semantic have closer distance than the dissimilar ones, and the positive-negative activity in the same video have large distance. To further filter out the background information in each video, the foreground attention branch is designed to distinguish the foreground frames. The details of our main grounding process is shown in Figure 2. During the training stage, we utilize the pseudo labels, which are obtained from the deep semantic features, as guidance to refine the video grounding model with an iterative learning procedure. To sum up, our main contributions are as follows:

- To the best of our knowledge, this is the first work to address temporal video grounding in the unsupervised setting. Without supervision, we solve the task with the proposed DSCNet, which learns to compose the activity contents guided by the deep language semantic.
- We use an encoder-decoder module to obtain the semantic features for all queries and divide them into different clusters to represent different semantic meanings. Then we propose a two-branch video module, where specific attention branch aggregates the query semantic to match the segment, and the foreground attention branch is utilized to distinguish the foreground-background activities.
- We conduct comprehensive experiments on the ActivityNet Captions and Charade-STA datasets. The results demonstrate the effectiveness of our proposed method, where DSCNet achieves decent results and outperforms most weakly-supervised methods.

Related Work

Fully-supervised temporal video grounding. Most of the existing methods refer to fully-supervised setting where all video-sentence pairs are labeled in details, including corresponding segment boundaries. Therefore, the main challenge in such setting is how to align multi-modal features well to predict precise boundary. Some works (Qu et al. 2020; Liu, Qu, and Zhou 2021; Liu et al. 2021a, 2020a, 2022b,a) integrate sentence information with each fine-grained video clip unit, and predict the scores of candidate segments by gradually merging the fusion feature sequence over time. Although these methods achieve good performances, they severely rely on the quality of the proposals and are time-consuming. Without using proposals, the latest methods (Nan et al. 2021; Mun, Cho, and Han 2020; Zeng et al. 2020) are proposed to leverage the interaction between video and sentence to predict the starting and ending frames directly. These methods are more efficient than the proposal-based ones, but achieve lower performance.

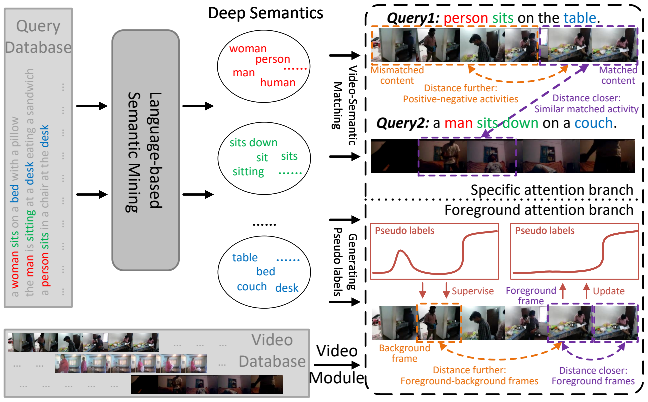

Query Database
a woman sits on a bed with a pillow
the man is sitting at a desk eating a sandwich
a person sits in a chair at the desk
Language-based Semantic Mining
Deep Semantics
woman person man human ......
sits down sit sits sitting ......
......
table bed couch desk ......
Video-Semantic Matching
Query1: person sits on the table.
Mismatched content
Matched content
Distance further: Positive-negative activities
Distance closer: Similar matched activity
Query2: a man sits down on a couch.
Specific attention branch
Foreground attention branch
Generating Pseudo labels
Pseudo labels
Pseudo labels
Supervise
Foreground frame
Update
Background frame
Distance further: Foreground-background frames
Distance closer: Foreground frames
Video Database
Video Module

Figure 2: The main idea of our proposed method, where we only show several semantic clusters for example.

Weakly-supervised temporal video grounding. To ease the human labelling efforts, several works (Bojanowski et al. 2015; Mithun, Paul, and Roy-Chowdhury 2019; Lin et al. 2020; Song et al. 2020; Zhang et al. 2020b; Ma et al. 2020; Tan et al. 2021) consider a weakly-supervised setting which only access the information of matched video-query pairs without accurate segment boundaries. (Mithun, Paul, and Roy-Chowdhury 2019) utilize the dependency between video and sentence as the supervision while abandon the temporal ordered information. Their text-guided attention provides scores for segment proposals. (Lin et al. 2020) put forward a module to reconstruct sentences and a proposal reward is based on the loss calculated using the target sentence and reconstructed one. Though these weakly-supervised methods do not rely on the temporal annotations, they still need the dependency between video and sentence as supervision. Different from them, we are the first to attempt to solve this task with an unsupervised approach that does not require any video-query dependency.

Unsupervised Learning. Recently, unsupervised methods (Soomro and Shah 2017; Laina, Rupprecht, and Navab 2019; Su, Zhong, and Zhang 2019; Gong et al. 2020) receive increasing attention in multi-modal retrieval task. Laina (Laina, Rupprecht, and Navab 2019) and Su (Su, Zhong, and Zhang 2019) embed both video and text into a shared latent space, then maximally reconstruct the joint-

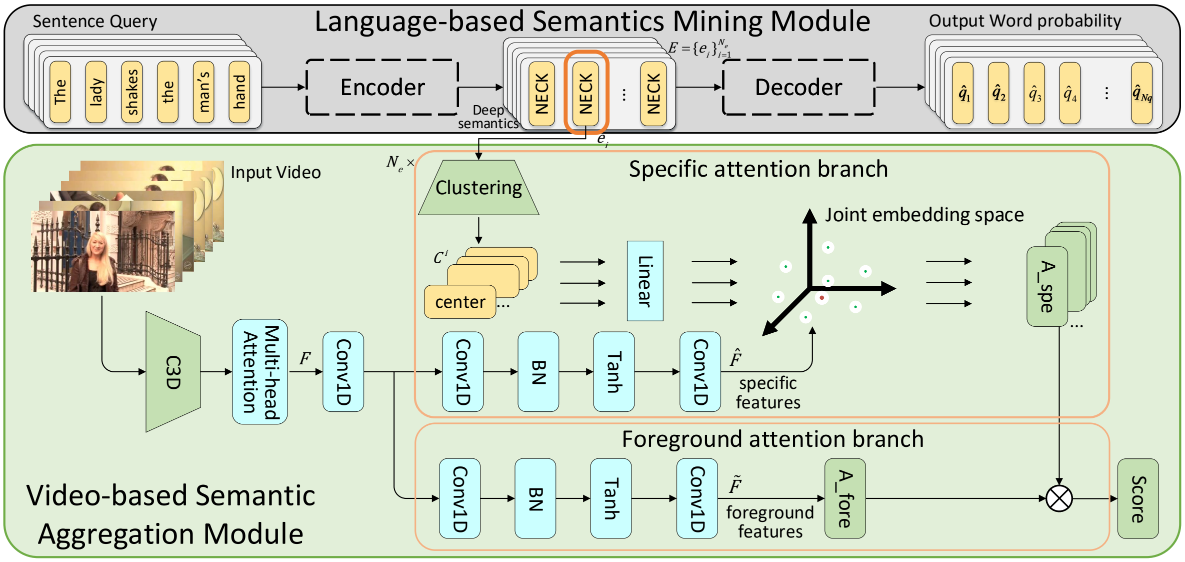


Figure 3: The overall architecture of the proposed DSCNet for unsupervised TVG task. Given a query set, we first develop a language-based semantic mining module to learn the deep semantic for all queries by an encoder-decoder model. Then a video-based semantic aggregation module is proposed to compose the possible activities referring to the deep semantic clusters.

semantics relations. Soomro (Soomro and Shah 2017) and Gong (Gong et al. 2020) propose unsupervised action localization and transform the task into a unsupervised frame-wise classification problem with the pre-defined action categories. Different from these retrieval methods, the unsupervised TVG task requires fine-grained video-query alignment for better predicting accurate start-end timestamps. To learn deep semantics of sentence queries, we utilize widely used encoder-decoder architecture (Rumelhart, Hinton, and Williams 1968; Olshausen and Field 1997; Vincent et al. 2008; Rifai et al. 2011; Kingma and Welling 2014) to learn its unsupervised representations, which consists of an encoder to extract feature representations and a decoder to reconstruct the input data from the representations. Then we compose the activity contents among the video guided by these learnt deep semantic features.

Deep Semantics Clustering Network

Preliminaries

In the TVG task, we are provided with a training set of untrimmed videos $\boldsymbol{V} = \{\boldsymbol{v}_i\}_{i=1}^{N}$ and sentence queries $\boldsymbol{Q} = \{\boldsymbol{q}_i\}_{i=1}^{M}$, where $\boldsymbol{v}_i$ and $\boldsymbol{q}_i$ are the i-th video and query, N and M are their corresponding numbers. Since our method is under unsupervised setting, we drop all label information between $\boldsymbol{V}$ and $\boldsymbol{Q}$ including both their correspondence and annotated segment boundaries.

The overall architecture of our proposed Deep Semantic Clustering Network (DSCNet) is shown in Figure 3. We first develop an independent encoder-decoder network to learn deep semantic features for the whole query set. In particular, we extract the hidden representations of each sentence as deep semantic called "neck", and gather the necks of all queries into different semantic clusters by a clustering algorithm. Furthermore, we devise a video-based aggregation model with two attention branches: specific attention branch and foreground attention branch. Specifically, the specific attention branch is devised to match the frame-wise feature with each semantic cluster, and the foreground attention branch is developed to distinguish the foreground-background events.

Language-based Semantic Mining

In this section, we make an attempt to extract the internal information of each query by reconstructing its main meaning with an independent encoder-decoder network. Meanwhile, a well-designed loss function $\mathcal{L}_w$ in Eq. (3) is proposed to learn the discriminative hidden feature of input queries, which is also named as deep semantic "neck" thereafter.

For encoder, given a sentence query, we first employ off-the-shelf Glove model (Pennington, Socher, and Manning. 2014) to obtain its word-level embedding $\boldsymbol{W} \in \mathbb{R}^{L\times d_w}$, where L is the sentence length and d_w is the embedding dimension. Then we feed $\boldsymbol{W}$ into a two-layer LSTM network and use the last hidden unit as the sentence-level representation $\boldsymbol{r}_e \in \mathbb{R}^{d_r}$. To separate different latent semantic representations which describe different aspects in $\boldsymbol{r}_e$, we further feed the sentence-level representation $\boldsymbol{r}_e$ into multiple two-layer perceptrons to obtain the hidden features of the encoder-decoder model, called "necks", which serve as the implicit semantics of queries. Specifically, we denote the necks of a query as $\boldsymbol{E} = \{\boldsymbol{e}_i\}_{i=1}^{N_e} \in \mathbb{R}^{N_e\times d_e}$, where N_e is the neck number and d_e is dimension.

To ensure that the learned necks have contained the most crucial information of the sentence, we further adopt a decoder module to reconstruct the original sentence with these necks. In details, we first aggregate the information from all necks to output a new sentence-level representation $\boldsymbol{r}_o \in \mathbb{R}^{d_r}$ by other multiple two-layer perceptrons with further concatenation. Then, we feed it into another two-layer LSTM network and a linear layer to construct the word-level sentence, which is expected to be the same as the input one. Specifically, for the i-th output word, the decoder outputs its score vector $\{p_{i,j}\}_{j=1}^{N_w}$, where N_w is the size of vocabulary in the whole query set, and $p_{i,j}$ means the probability distribution of j-th word in the vocabulary appearing at the i-th output word. Suppose the predicted probabilities of the word-level groundtruth location in original L-length input sentence as $\{p_{i,truth}\}_{i=1}^{L}$, we calculate Cross Entropy Loss $\mathcal{L}_{cel}$ with softmax function to supervise the output sentence as:

$$\mathcal{L}_{cel} = -\sum_{i=1}^{L} log(\frac{exp(p_{i,truth})}{\sum_{j=1}^{N_w} exp(p_{i,j})}). \tag{1}$$

Besides, to ensure both the input and output sentences have the same sentence-level semantic meaning, we add a semantic loss $\mathcal{L}_{mse}$ calculated by the Mean Square Error function between both sentence-level representations $\boldsymbol{r}_e$ and $\boldsymbol{r}_o$. More importantly, since we expect each neck in the query has unique semantic, we adopt a regularization term (Lin et al. 2017) $\mathcal{L}_{dqa}$ to enforce necks $\{\boldsymbol{e_i}\}_{i=1}^{N_e}$ be different from each other:

$$\mathcal{L}_{dqa} = \|(\boldsymbol{E}^{\top}\boldsymbol{E}) - \lambda \boldsymbol{I}\|, \tag{2}$$

where $\|\cdot\|$ denotes L2-norm, $\lambda \in (0, 1]$ controls the extent of overlap between different necks. $\boldsymbol{I}$ is an identity matrix. By combining the above three losses with balanced parameters α_w, β_w, we can adopt a multi-task loss function in the semantic mining module as follows:

$$\mathcal{L}_w = \mathcal{L}_{cel} + \alpha_w \mathcal{L}_{mse} + \beta_w \mathcal{L}_{dqa}. \tag{3}$$

In a similar way, we can get the neck features for all sentences in $\boldsymbol{Q}$, where each sentence has N_e necks. Subsequently, for the i-th neck of all queries, we implement K-means clustering algorithm (Na, Xumin, and Yong 2010) to get N_c centers upon them. Formally, the centers of the i-th necks are recorded as $\boldsymbol{C}^i = \{\boldsymbol{c}_j^i\}_{j=1}^{N_c} \in \mathbb{R}^{N_c \times d_e}$, d_e is the dimension of each center feature. These centers can be regarded as discriminative semantic features. In the following video-based semantic aggregation module, such central semantic representations can be utilized for activity content composing.

Video-based Semantic Aggregation

To compose possible activities referring to the central semantic features from the queries, we develop a video-based semantic aggregation module which consists of a specific attention branch and a foreground attention branch. During the training of the video module, we initialize a pseudo label for video frames as weak guidance to assist grounding, and utilize an iterative learning strategy to update and refine the pseudo labels for better training.

Video feature encoding. Given a video, we first utilize a C3D (Tran et al. 2015) network with a multi-head self-attention module (Vaswani et al. 2017) to extract the frame-wise features as $\boldsymbol{F} = \{\boldsymbol{f}_t\}_{t=1}^{T} \in \mathbb{R}^{T \times d_v}$, where T is the number of frames in one video and d_v is the channel dimension of frame-wise representation.

Pseudo labels. For specific semantic cluster $\boldsymbol{C}^i$, we initialize the pseudo labels $\boldsymbol{Y} = \{\boldsymbol{y}_j\}_{j=1}^{N_c} \in \mathbb{R}^{N_c \times T}$ to all T frames, where each label denotes whether a specific frame t matches the semantic cluster center $\boldsymbol{c}_j^i$. Specifically, we implement N-cut (Shi and Malik 2000) clustering with Gaussian kernel upon the concatenated feature $[\boldsymbol{f}_t; \boldsymbol{c}_j^i]$ to assign the binary label $\boldsymbol{y}_j \in \mathbb{R}^T$ label for each frame. Such clustering process can obtain coarse activity information. The fine-grained label would be obtained through the iterative learning process.

Specific attention branch. In this branch, we aim to aggregate different semantics of the query set for better composing possible activities among the video. After getting the semantic centers $\boldsymbol{C}^i = \{\boldsymbol{c}_j^i\}_{j=1}^{N_c}$ of i-th neck of the whole query set, we first project both language and video features into a joint embedding space, and denote their new features as $\hat{\boldsymbol{C}}^i \in \mathbb{R}^{N_c \times d_{e'}}$ and $\hat{\boldsymbol{F}} = \{\hat{\boldsymbol{f}}_t\}_{t=1}^{T} \in \mathbb{R}^{T \times d_{e'}}$, respectively. Then, we calculate the correlations between all frame-semantic pairs as the specific attention matrix $\boldsymbol{A}_{spe}$:

$$\boldsymbol{A}_{spe} = Softmax(\hat{\boldsymbol{C}}^i(\hat{\boldsymbol{F}})^{\top}) \in \mathbb{R}^{N_c \times T}, \tag{4}$$

where each row of $\boldsymbol{A}_{spe}$ denotes the similarities of all frames to a specific semantic center, and those frames with the corresponding highest scores will be composed into the activities.

In general, given a batch of training videos, we randomly sample J semantic cluster centers and Z videos. For v-th video feature $\hat{\boldsymbol{F}}_v$, we can calculate its specific positive activity features guided by j-th semantic cluster center as:

$$\tilde{\boldsymbol{S}}_{v,j}^{p} = \boldsymbol{A}_{spe}[j, :]\hat{\boldsymbol{F}}_v \in \mathbb{R}^{1 \times d_{e'}}. \tag{5}$$

We can also generate its negative specific feature by:

$$\boldsymbol{B}_{spe} = \frac{\boldsymbol{1} - \boldsymbol{A}_{spe}}{T}, \tag{6}$$

$$\tilde{\boldsymbol{S}}_{v,j}^{n} = \boldsymbol{B}_{spe}[j, :]\hat{\boldsymbol{F}}_v \in \mathbb{R}^{1 \times d_{e'}}, \tag{7}$$

where $\boldsymbol{1}$ here is a matrix with the same shape as $\boldsymbol{A}_{spe}$ filled by integer 1. Dividing by T is for the purpose of normalization. Since we expect the integrated positive features (different videos v, u about the same semantic center j) having the same semantic information to be similar while the positive and negative features of the same video (video v) to be distinct, our loss function $\mathcal{L}_{sab}$ of specific attention branch can be formulated as follows where $d(\cdot)$ is the cosine distance:

$$\mathcal{L}_{sim}^{(j,v)} = \sum_{u=1, u \neq v}^{Z} max[d(\tilde{\boldsymbol{S}}_{v,j}^{p}, \tilde{\boldsymbol{S}}_{u,j}^{p}) - \tau_1, 0], \tag{8}$$

$$\mathcal{L}_{dis}^{(j,v)} = \sum_{u=1, u \neq v}^{Z} max[d(\tilde{\boldsymbol{S}}_{v,j}^{p}, \tilde{\boldsymbol{S}}_{u,j}^{p}) - d(\tilde{\boldsymbol{S}}_{v,j}^{p}, \tilde{\boldsymbol{S}}_{v,j}^{n}) + \tau_2, 0], \tag{9}$$

$$\mathcal{L}_{sab} = \sum_{j=1}^{J}\sum_{v=1}^{Z}(\mathcal{L}_{sim}^{(j,v)} + \theta \mathcal{L}_{dis}^{(j,v)}), \tag{10}$$

where τ_1 and τ_2 denote margins, θ is a weight coefficient. In this way, the specific attention branch can aggregate most relevant frame-wise features corresponding to the specific semantic cluster center for activity composing.

Foreground attention branch. Only using specific attention is not enough to provide accurate grounding, as it can not filter out all background activities composed by the irrelevant semantics features. Therefore, as shown in Figure 3, we design a foreground attention branch with foreground features $\tilde{\boldsymbol{F}}$ and softmax attention output $\boldsymbol{A}_{fore} \in \mathbb{R}^{T\times 1}$, which can be obtained with several CNN layers. We first develop a triplet loss to distinguish the foreground-background frame-wise features according to the learned pseudo labels, where we aim to pull the frame representations of foreground (frames u, v) closer and push the frame representations of foreground-background (frames u, o) further in the feature space. Specifically, frame v is the foreground frame which has the minimum distance to u, and frame o is the background frame which has the maximum distance to u. The triple loss function can be formulated as:

$$\mathcal{L}_{trip} = \sum_{u=1}^{Z} max[d(\tilde{\boldsymbol{f}}_u^p, \tilde{\boldsymbol{f}}_v^p) - d(\tilde{\boldsymbol{f}}_u^p, \tilde{\boldsymbol{f}}_o^b) + \tau_3, 0]. \quad (11)$$

To predict the matching score of each frame by learning both specific attention matrix $\boldsymbol{A}_{spe} \in \mathbb{R}^{N_c\times T}$ and foreground attention matrix $\boldsymbol{A}_{fore} \in \mathbb{R}^{T\times 1}$ with the supervision of pseudo labels $\boldsymbol{Y} \in \mathbb{R}^{T\times N_c}$, we first calculate the score h_j^t of t-th frame referring to semantic center j and the foreground attention matrix $\boldsymbol{A}_{fore}$, and then formulate the grounding loss $\mathcal{L}_{cls}^t$ of each frame t as:

$$h_j^t = \boldsymbol{A}_{spe}^{\top}[t, j] \times \boldsymbol{A}_{fore}[t, :], \quad (12)$$

$$\mathcal{L}_{cls}^t = \sum_{j=1}^{N_c} [\boldsymbol{Y}[t, j] log(h_j^t) + (1 - \boldsymbol{Y}[t, j]) log(1 - h_j^t)] \quad (13)$$

where $\boldsymbol{A}_{spe}[t, j]$ denotes whether the frame t is the positive frame referring to the semantic center j, and $\boldsymbol{A}_{fore}[t, :]$ is utilized to determine whether the frame is the foreground. Then, we calculate the grounding loss for whole frames as:

$$\mathcal{L}_{cls} = -\sum_{v=1}^{Z}\sum_{t=1}^{T} \mathcal{L}_{cls}^t. \quad (14)$$

Combining the aforementioned three losses in two attention branches, we get the overall multi-task loss $\mathcal{L}_v$ in the video grounding model as:

$$\mathcal{L}_v = \mathcal{L}_{cls} + \alpha_v \mathcal{L}_{sab} + \beta_v \mathcal{L}_{trip}. \quad (15)$$

where α_v and β_v are hyper-parameters.

Iterative learning. We use an iterative optimization strategy to train our video module. In each iteration: (1) we update the pseudo label on each frame by applying the cluster algorithm on the new feature $[\hat{\boldsymbol{f}}_t; \boldsymbol{c}_j^i]$, where $\hat{\boldsymbol{f}}_t$ is the learned feature in the specific branch as it contains more semantic-aware contexts. (2) we calculate and back-propagate the loss $\mathcal{L}_v$ for updating the video model. The overall training process is shown in Algorithm 1. During the iterative training, the grounding module gradually finds the important frames of the video and yields a better frame-wise feature representation. Such precise feature representations can further lead to more precise pseudo labels obtained from the clustering process, and in turn provides better supervisions for the grounding. We show the effectiveness of the iterative learning process in our experiments.

Algorithm 1 Iterative learning process of video module

Input: All semantic cluster centers $\boldsymbol{C}$ of the whole query set; video feature $\boldsymbol{F}$.

```
Init pseudo label based on $\boldsymbol{C}$ and $\boldsymbol{F}$
for iteration $l \leftarrow 1$ to $L$ do
    for neck $i \leftarrow 1$ to $N_e$ do
        Execute specific attention branch with $\boldsymbol{C}^i = \{\boldsymbol{c}_j^i\}_{j=1}^{N_c}$ to obtain $\hat{\boldsymbol{F}}_v = \{\hat{\boldsymbol{f}}_v^t\}_{t=1}^{T}$ and $\boldsymbol{A}_{spe}$;
        Execute foreground attention branch to obtain $\boldsymbol{A}_{fore}$;
        Generate the training samples by pseudo labels, and calculate the overall loss $\mathcal{L}_v$ for back-propagation;
        Generate the new feature $\hat{\boldsymbol{F}}_v$, and utilize it to update the pseudo labels;
    end
end
```

Inference

When testing, we directly utilize the generated N_e necks feature $\boldsymbol{E}$ of the input query as semantic cluster centers, and feed them into the video module to match with frame-wise features for generating corresponding specific attention $\boldsymbol{A}_{spe} \in \mathbb{R}^{N_e\times T}$ and foreground attention $\boldsymbol{A}_{fore} \in \mathbb{R}^{1\times T}$ for activity composing. Specifically, we element-wisely multiply $\boldsymbol{A}_{spe}$ and $\boldsymbol{A}_{fore}$ for each neck, and feed the results of all necks to softmax layers with a further element-wise multiplication. The final attention scores of size $1 \times T$ denotes a joint probability of all semantics. Finally, we locate the frame with the highest score as the basic predicted segment, and add the left/right frames into the segment if the ratio of their scores to the frame score of the closest segment boundary is less than a threshold. We repeat this step until no frame can be added.

Experimental Results

Datasets and Evaluation

Charades-STA. This dataset is built from the Charades (Sigurdsson et al. 2016) dataset and transformed into video temporal grounding task by (Gao et al. 2017). It contains 16128 video-sentence pairs with 12408 pairs used for training and 3720 for testing. The videos are about 30 seconds on average. The annotations are generated by sentence decomposition and keyword matching with manually check.

ActivityNet Captions. This dataset is built from ActivityNet v1.3 dataset (Caba Heilbron et al. 2015) for dense video captioning. It contains 20000 YouTube videos with 100000 queries. We follow the public split of the dataset that contains a training set and two validation sets val 1 and val 2. On average, videos are about 120 seconds and queries are about 13.5 words.

Evaluation. Following prior work (Gao et al. 2017), we adopt "R@N, IoU=θ" as our evaluation metrics, which is defined as the percentage of at least one of top-N selected moments having IoU scores larger than θ.

Methods	Mode	Charades-STA						ActivityNet Captions			
		R@1	R@1	R@1	R@5	R@5	R@5	R@1	R@1	R@5	R@5
		IoU=0.3	IoU=0.5	IoU=0.7	IoU=0.3	IoU=0.5	IoU=0.7	IoU=0.3	IoU=0.5	IoU=0.3	IoU=0.5
Rondom	FS	20.12	8.51	3.03	68.42	37.12	14.06	18.64	7.73	52.78	29.49
VSA-RNN	FS	-	10.50	4.32	-	48.43	20.21	39.28	24.43	70.84	55.52
VSA-STV	FS	-	16.91	5.81	-	53.89	23.58	41.71	24.01	71.05	56.62
CTRL	FS	-	23.62	8.89	-	58.92	29.52	-	-	-	-
TGN	FS	-	-	-	-	-	-	43.81	27.93	54.56	44.20
EFRC	FS	53.00	33.80	15.00	94.60	77.30	43.90	-	-	-	-
2D-TAN	FS	-	39.81	23.25	-	79.33	52.15	59.45	44.51	85.53	77.13
DRN	FS	-	45.40	26.40	-	88.01	55.38	-	45.45	-	77.97
TGA	WS	32.14	19.94	8.84	86.58	65.52	33.51	-	-	-	-
SCN	WS	42.96	23.58	9.97	95.56	71.80	38.87	47.23	29.22	71.45	55.69
CTF	WS	39.80	27.30	12.90	-	-	-	44.30	23.60	-	-
MARN	WS	48.55	31.94	14.18	90.70	70.00	37.40	47.01	29.95	72.02	57.49
VGN	WS	-	32.21	15.68	-	73.50	41.87	50.12	31.07	77.36	61.29
DSCNet	US	**44.15**	**28.73**	**14.67**	**91.48**	**70.68**	**35.19**	**47.29**	**28.16**	**72.51**	**57.24**

Table 1: Performance comparisons for video grounding on both Charades-STA and ActivityNet Captions datasets, where FS: fully-supervised setting, WS: weakly-supervised setting and US: unsupervised setting.

Method	Language Module			Video Module			R@1	R@1	R@1	R@5	R@5	R@5
	$\mathcal{L}_{cel}$	$\mathcal{L}_{dqa}$	$\mathcal{L}_{mse}$	$\mathcal{L}_{sab}$	$\mathcal{L}_{trip}$	$\mathcal{L}_{cls}$	IoU=0.3	IoU=0.5	IoU=0.7	IoU=0.3	IoU=0.5	IoU=0.7
baseline	✓	×	×	×	×	✓	28.38	16.12	7.86	76.18	48.65	20.21
+ $\mathcal{L}_{dqa}$	✓	✓	×	×	×	✓	33.54	20.07	9.09	80.16	55.41	23.72
+ $\mathcal{L}_{w}$	✓	✓	✓	×	×	✓	36.78	23.14	10.65	82.98	60.29	26.31
+ $\mathcal{L}_{w}$ + $\mathcal{L}_{trip}$+ $\mathcal{L}_{cls}$	✓	✓	✓	×	✓	✓	40.23	25.32	12.11	86.83	65.74	30.45
+ $\mathcal{L}_{w}$ + $\mathcal{L}_{sab}$+ $\mathcal{L}_{cls}$	✓	✓	✓	✓	×	✓	41.97	26.01	12.39	87.61	67.33	32.60
+ $\mathcal{L}_{w}$ + $\mathcal{L}_{v}$	✓	✓	✓	✓	✓	✓	**44.15**	**28.73**	**14.67**	**91.48**	**70.68**	**35.19**

Table 2: The analysis of each component in DSCNet, via ablation study on Charades-STA.

Implementation Details

In order to make a fair comparison with previous works, we utilize C3D to extract video features and Glove to obtain word embeddings. As some videos are too long, we set the length of video feature sequences to 128 for Charades-STA and 256 for ActivityNet Captions. We fix the query length to 10 in Charades-STA and 20 in ActivityNet Captions. We set neck number N_e to 4 for Charades-STA and 8 for ActivityNet Captions, and set cluster number N_c to 16. The LSTM Layers in language encoder and decoder are both 2 layers architecture with 512 hidden size. The dimension of joint embedding space $d_{e'}$ is set to 1024. We utilize Adam optimizer with the initial learning rate as 0.0001 for language module and 0.0005 for video module. The hyper-parameters $\theta, \tau_1, \tau_2, \tau_3$ are set as 1.0, 0.0001, 0.0001, 0.5. λ is set to 0.5. And $\alpha_w, \beta_w, \alpha_v, \beta_v$ in Eq. (3) and (15) are all set as 0.5. The inference threshold is set to 0.8 in ActivityNet and 0.9 in Charades-STA.

Comparisons with state-of-the-arts

Comparison on Charades-STA. We first compare our model DSCNet with the state-of-the-art methods on Charades-STA dataset, shown in Table 1. Specifically, for metrics R@1, IoU∈{0.3,0.5,0.7}, the results achieved by our method in the unsupervised setting (US) are comparable to the results obtained by the state-of-the-art fully-supervised (FS) and weakly-supervised (WS) methods. For R@5, we also have similar observations.

Comparison on ActivityNet Captions. We also presents the results on ActivityNet Captions, shown in Table 1. We compare our method DSCNet with other recent state-of-the-art FS and WS video grounding methods. Even without any video-query annotations, our method is able to achieve 47.29%, 28.16%, 72.51%, 57.24% in all metrics, showing competitive performance comparing to almost all weakly supervised methods.

Ablation Study

In this section, we conduct ablation study to validate the effectiveness of each components in our methods. All experiments are conducted on Charades-STA dataset.

Effect of each component. To analyze how each model component contributes to the task, we perform ablation study as shown in Table 2. We use the model in which the language-based encoder-decoder is only combined with reconstruction loss $\mathcal{L}_{cel}$, and the video module is combined with grounding loss $\mathcal{L}_{cls}$ as our baseline. Then we add the regularization loss $\mathcal{L}_{dqa}$ to the baseline model and improve R@1 IoU=0.3 from 28.38% to 33.54%, R@1 IoU=0.5 from 16.12% to 20.07%, demonstrating the importance of learning different semantic features. The sentence-level semantic loss $\mathcal{L}_{mse}$ can further improve the hidden features learning of the auto-encoder model. The special attention branch loss $\mathcal{L}_{sab}$ and triple loss $\mathcal{L}_{trip}$ also bring significant improvements by yielding better frame-wise representations. From the table, we can see that jointly combining all the loss functions achieves the superior overall performance.

How to select the cluster center. As shown in Table 3, we investigate how the strategy of selecting the center $\boldsymbol{C}$ affects the grounding results. We can find that randomly selection

Methods	R@1 IoU=0.3	R@1 IoU=0.5	R@5 IoU=0.3	R@5 IoU=0.5
random	23.79	11.24	73.61	42.58
cluster sample	39.67	23.96	86.13	66.49
cluster center	**44.15**	**28.73**	**91.48**	**70.68**

Table 3: Ablation study on the selection of the semantic cluster center $\boldsymbol{C}$.

Methods	R@1 IoU=0.3	R@1 IoU=0.5	R@5 IoU=0.3	R@5 IoU=0.5
2 necks	33.06	19.70	79.39	53.53
4 necks	**44.15**	**28.73**	**91.48**	**70.68**
8 necks	40.90	25.03	87.56	65.40
1 iteration	29.38	15.61	81.13	60.08
3 iteration	39.98	24.87	88.09	66.77
5 iteration	**44.15**	**28.73**	91.48	**70.68**
7 iteration	44.06	28.39	**91.82**	70.41

Table 4: Ablation study on the neck number and iteration number on Charades-STA dataset.

without clustering performs poorly, since it lacks sufficient semantics to compose all possible activities. The "cluster center" (we directly choose the averaged cluster center) performs much better than the "cluster sample", where we randomly choose one sample in each cluster. The reason could be the central embedding in each cluster contains the most representative semantic to cover other samples.

Comparing different neck numbers. For the number of necks, as shown in Table 4, using 2 necks which contains complex information leads to poor scores. Using 4 necks achieves the best performance while using 8 necks is slightly lower. Therefore, we choose 4 as the neck number on Charades-STA dataset in our experiments.

Comparing different iterative learning times. Table 4 also show the performance on different iterations. As the number of iterations increases, the performance becomes better. Our model achieves the best results with 5 iterations. We do not see more improvements by increasing iterations after 5.

Visualization Results

Figure 4 shows some grounding results of our DSCNet on Charades-STA dataset. Figure 5 shows the semantic clustering results of the language module, in which we partially visualize the clusters related to actions and objects by utilizing T-SNE (van der Maaten and Hinton 2008). We select sentences containing specific words from the test set of Charades-STA for visualization. For class actions, we take "drink", "eat", "run", "walk" as the examples. Through clustering, the actions "drink", "eat" are quite different from actions "run", "walk" since they generally appear in different scenarios (indoor vs. outdoor). Meanwhile, we can observe that there is a distinct margin between the "drink", "eat" and "run", "walk" as shown in the left figure. Furthermore, we can also find that actions "drink" and "eat" are well separated while "run" and "walk" are not, this is because: actions "run" and "walk" are quite close in semantics, and can be substituted for each other in some circumstances. The right figure shows the clustering results on multiple objects. It illustrates that the objects "cup", "glass" are the intra-pairs, and the objects "window", "door" are the inter-pairs.

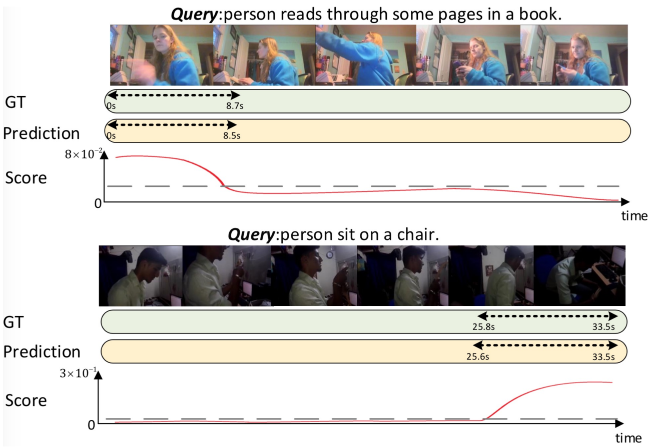


Figure 4: Qualitative results on charades-STA dataset. "GT" is the annotation of the ground-truth segment and "Prediction" is our grounding result. The score value in the red curve of each video denotes the probability of each frame.

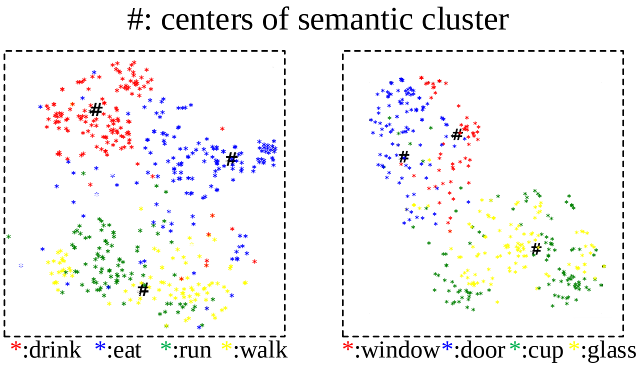


Figure 5: Example of the semantic clustering results of the language-based semantic mining module.

Conclusion

In this paper, we propose a novel Deep Semantic Clustering Network (DSCNet), to solve the temporal video grounding (TVG) task under the unsupervised setting. We first mine deep semantic features from all sentences and apply clustering on them to obtain the universal textual representation of the whole query set. Then, we compose the possible activities among the videos guided by the extracted deep semantic features. Specifically, we design two attention branches with the novel loss function for grounding. Our method is evaluated on two benchmark datasets and achieves decent performances, compared with most fully/weekly supervised baselines. The future work includes applying DSCNet to other tasks/datasets (Li et al. 2020; Lei et al. 2020), and leveraging local/global features to learn better video-text representations. Following the idea of DSCNet, we would like to explore how to use more unannotated data in supervised manner.

Acknowledgements

This work was supported in part by the National Natural Science Foundation of China (NSFC) under Grant (No.61972448, No.62172068, No.61802048).

References

Bojanowski, P.; Lajugie, R.; Grave, E.; Bach, F.; Laptev, I.; Ponce, J.; and Schmid, C. 2015. Weakly-supervised alignment of video with text. *Proceedings of the IEEE International Conference on Computer Vision (ICCV)* .

Caba Heilbron, F.; Escorcia, V.; Ghanem, B.; and Carlos Niebles, J. 2015. ActivityNet: A Large-Scale Video Benchmark for Human Activity Understanding. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* .

Chen, J.; Ma, L.; Chen, X.; Jie, Z.; and Luo, J. 2019. Localizing natural language in videos. *Proceedings of the American Association for Artificial Intelligence* .

Chen, L.; Lu, C.; Tang, S.; Xiao, J.; Zhang, D.; Tan, C.; and Li, X. 2020. Rethinking the bottom-up framework for query-based video localization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, 10551–10558.

Gao, J.; Sun, C.; Yang, Z.; and Nevatia, R. 2017. Tall: Temporal activity localization via language query. *Proceedings of the IEEE International Conference on Computer Vision (ICCV)* 5267–5275.

Gong, G.; Wang, X.; Mu, Y.; and Tian, Q. 2020. Learning Temporal Co-Attention Models for Unsupervised Video Action Localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 9819–9828.

Kingma, D. P.; and Welling, M. 2014. Auto-Encoding Variational Bayes. *arXiv:1312.6114* .

Laina, I.; Rupprecht, C.; and Navab, N. 2019. Towards unsupervised image captioning with shared multimodal embeddings. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 7414–7424.

Lei, J.; Yu, L.; Berg, T. L.; and Bansal, M. 2020. TVR: A Large-Scale Dataset for Video-Subtitle Moment Retrieval. In *Proceedings of the European Conference on Computer Vision (ECCV).*

Li, L.; Chen, Y.-C.; Cheng, Y.; Gan, Z.; Yu, L.; and Liu, J. 2020. HERO: Hierarchical Encoder for Video+ Language Omni-representation Pre-training. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP).*

Lin, Z.; Feng, M.; Nogueira dos Santos, C.; Yu, M.; Xiang, B.; Zhou, B.; and Bengio, Y. 2017. A Structured Self-Attentive Sentence Embedding. *International Conference on Learning Representations(ICLR)* .

Lin, Z.; Zhao, Z.; Zhang, Z.; Wang, Q.; and Liu, H. 2020. Weakly-supervised video moment retrieval via semantic completion network. *Proceedings of the American Association for Artificial Intelligence* .

Liu, D.; Qu, X.; Di, X.; Cheng, Y.; Xu, Z. X.; and Zhou, P. 2022a. Memory-Guided Semantic Learning Network for Temporal Sentence Grounding. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Liu, D.; Qu, X.; Dong, J.; and Zhou, P. 2020a. Reasoning step-by-step: Temporal sentence localization in videos via deep rectification-modulation network. In *Proceedings of the 28th International Conference on Computational Linguistics*, 1841–1851.

Liu, D.; Qu, X.; Dong, J.; and Zhou, P. 2021a. Adaptive Proposal Generation Network for Temporal Sentence Localization in Videos. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 9292–9301.

Liu, D.; Qu, X.; Dong, J.; Zhou, P.; Cheng, Y.; Wei, W.; Xu, Z.; and Xie, Y. 2021b. Context-aware Biaffine Localizing Network for Temporal Sentence Grounding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 11235–11244.

Liu, D.; Qu, X.; Liu, X.-Y.; Dong, J.; Zhou, P.; and Xu, Z. 2020b. Jointly cross-and self-modal graph attention network for query-based moment localization. In *Proceedings of the 28th ACM International Conference on Multimedia*, 4070–4078.

Liu, D.; Qu, X.; and Zhou, P. 2021. Progressively Guide to Attend: An Iterative Alignment Framework for Temporal Sentence Grounding. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 9302–9311.

Liu, D.; Qu, X.; Zhou, P.; and Liu, Y. 2022b. Exploring Motion and Appearance Information for Temporal Sentence Grounding. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Ma, M.; Yoon, S.; Kim, J.; Lee, Y.; Kang, S.; and Yoo, C. D. 2020. Vlanet: Video-language alignment network for weakly-supervised video moment retrieval. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 156–171.

Mithun, N. C.; Paul, S.; and Roy-Chowdhury, A. K. 2019. Weakly supervised video moment retrieval from text queries. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* 11592–11601.

Mun, J.; Cho, M.; and Han, B. 2020. Local-Global Video-Text Interactions for Temporal Grounding. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* .

Na, S.; Xumin, L.; and Yong, G. 2010. Research on k-means clustering algorithm: An improved k-means clustering algorithm. In *2010 Third International Symposium on intelligent information technology and security informatics*, 63–67.

Nan, G.; Qiao, R.; Xiao, Y.; Liu, J.; Leng, S.; Zhang, H.; and Lu, W. 2021. Interventional Video Grounding with Dual Contrastive Learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2765–2775.

Olshausen, B. A.; and Field, D. J. 1997. Sparse coding with an overcomplete basis set: A strategy employed by V1? *Vision Research* 37(23): 3311–3325.

Pennington, J.; Socher, R.; and Manning., C. D. 2014. Glove: Global vectors for word representation. *Empirical Methods in Natural Language Processing (EMNLP)* 1532–1543.

Qu, X.; Tang, P.; Zou, Z.; Cheng, Y.; Dong, J.; Zhou, P.; and Xu, Z. 2020. Fine-grained iterative attention network for temporal language localization in videos. In *Proceedings of the 28th ACM International Conference on Multimedia*, 4280–4288.

Rifai, S.; Vincent, P.; Muller, X.; Glorot, X.; and Bengio, Y. 2011. Contractive Auto-Encoders: Explicit Invariance During Feature Extraction. *Proceedings of the International Conference on Machine Learning (ICML)* .

Rumelhart, D. E.; Hinton, G. E.; and Williams, R. J. 1968. Learning representations by back-propagating errors. *Nature* 533–536.

Shi, J.; and Malik, J. 2000. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22(8): 888–905.

Sigurdsson, G. A.; Varol, G.; Wang, X.; Farhadi, A.; Laptev, I.; and Gupta, A. 2016. Hollywood in homes: Crowdsourcing data collection for activity understanding. *Proceedings of the European Conference on Computer Vision (ECCV)* .

Singha, J.; Roy, A.; and Laskar, R. H. 2018. Dynamic hand gesture recognition using vision-based approach for human–computer interaction. *Neural Computing and Applications* 29(4): 1129–1141.

Song, Y.; Wang, J.; Ma, L.; Yu, Z.; and Yu, J. 2020. Weakly-supervised multi-level attentional reconstruction network for grounding textual queries in videos. *arXiv preprint arXiv:2003.07048* .

Soomro, K.; and Shah, M. 2017. Unsupervised action discovery and localization in videos. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 696–705.

Su, S.; Zhong, Z.; and Zhang, C. 2019. Deep joint-semantics reconstructing hashing for large-scale unsupervised cross-modal retrieval. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 3027–3035.

Tan, R.; Xu, H.; Saenko, K.; and Plummer, B. A. 2021. Logan: Latent graph co-attention network for weakly-supervised video moment retrieval. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2083–2092.

Tran, D.; Bourdev, L.; Fergus, R.; Torresani, L.; and Paluri, M. 2015. Learning spatiotemporal features with 3d convolutional networks. *Proceedings of the IEEE International Conference on Computer Vision (ICCV)* 4489–4497.

van der Maaten, L.; and Hinton, G. 2008. Visualizing Data using t-SNE. *Journal of Machine Learning Research* 9(86): 2579–2605.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is All you Need. *Advances in Neural Information Processing Systems (NIPS)* 30: 5998–6008.

Vincent, P.; Larochelle, H.; Bengio, Y.; and Manzagol, P.-A. 2008. Extracting and composing robust features with denoising autoencoders. *Proceedings of the International Conference on Machine Learning (ICML)* 1096–1103.

Wang, J.; Ma, L.; and Jiang, W. 2020a. Temporally grounding language queries in videos by contextual boundary-aware prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, 12168–12175.

Wang, J.; Ma, L.; and Jiang, W. 2020b. Temporally Grounding Language Queries in Videos by Contextual Boundary-aware Prediction. *Proceedings of the AAAI Conference on Artificial Intelligence* .

Zeng, R.; Xu, H.; Huang, W.; Chen, P.; Tan, M.; and Gan, C. 2020. Dense Regression Network for Video Grounding. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* .

Zhang, D.; Dai, X.; Wang, X.; Wang, Y.-F.; and Davis, L. S. 2019. Man: Moment alignment network for natural language moment retrieval via iterative graph adjustment. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1247–1257.

Zhang, S.; Peng, H.; Fu, J.; and Luo, J. 2020a. Learning 2D Temporal Adjacent Networks for Moment Localization with Natural Language. *Proceedings of the AAAI Conference on Artificial Intelligence* 34(07): 12870–12877.

Zhang, Z.; Lin, Z.; Zhao, Z.; Zhu, J.; and He, X. 2020b. Regularized Two-Branch Proposal Networks for Weakly-Supervised Moment Retrieval in Videos. In *Proceedings of the 28th ACM International Conference on Multimedia*, 4098–4106.

Zhao, Y.; Xiong, Y.; Wang, L.; Wu, Z.; Tang, X.; and Lin, D. 2017. Temporal action detection with structured segment networks. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2914–2923.