

Video Corpus Moment Retrieval with Contrastive Learning

Hao Zhang, Aixin Sun[†]
Nanyang Technological University,
Singapore
{hao007@e.,axsun@}ntu.edu.sg

Wei Jing
Institute of Infocomm Research,
A*STAR, Singapore
21wjing@gmail.com

Guoshun Nan
Singapore University of Technology
and Design, Singapore
nanguoshun@gmail.com

Liangli Zhen
Institute of High Performance
Computing, A*STAR, Singapore
zhenll@ihpc.a-star.edu.sg

Joey Tianyi Zhou
Institute of High Performance
Computing, A*STAR, Singapore
zhouty@ihpc.a-star.edu.sg

Rick Siow Mong Goh
Institute of High Performance
Computing, A*STAR, Singapore
gohsm@ihpc.a-star.edu.sg

ABSTRACT

Given a collection of untrimmed and unsegmented videos, *video corpus moment retrieval* (VCMR) is to retrieve a temporal moment (*i.e.*, a fraction of a video) that semantically corresponds to a given text query. As video and text are from two distinct feature spaces, there are two general approaches to address VCMR: (i) to separately encode each modality representations, then align the two modality representations for query processing, and (ii) to adopt fine-grained cross-modal interaction to learn multi-modal representations for query processing. While the second approach often leads to better retrieval accuracy, the first approach is far more efficient. In this paper, we propose a **Retrieval and Localization Network with Contrastive Learning** (ReLoCLNet) for VCMR. We adopt the first approach and introduce two contrastive learning objectives to refine video encoder and text encoder to learn video and text representations separately but with better alignment for VCMR. The video contrastive learning (VideoCL) is to maximize mutual information between query and candidate video at video-level. The frame contrastive learning (FrameCL) aims to highlight the moment region corresponds to the query at frame-level, within a video. Experimental results show that, although ReLoCLNet encodes text and video separately for efficiency, its retrieval accuracy is comparable with baselines adopting cross-modal interaction learning.¹

CCS CONCEPTS

• **Information systems** → **Video search**; **Novelty in information retrieval**.

KEYWORDS

Moment Localization; Temporal Video Grounding; Video Corpus Moment Retrieval; Cross-modal Retrieval; Contrastive Learning

[†]Aixin Sun is the corresponding author.

¹<https://github.com/IsaacChanghau/ReLoCLNet>

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

SIGIR '21, July 11–15, 2021, Virtual Event, Canada

© 2021 Copyright held by the owner/author(s). Publication rights licensed to ACM.

ACM ISBN 978-1-4503-8037-9/21/07...\$15.00

<https://doi.org/10.1145/3404835.3462874>

ACM Reference Format:

Hao Zhang, Aixin Sun, Wei Jing, Guoshun Nan, Liangli Zhen, Joey Tianyi Zhou, and Rick Siow Mong Goh. 2021. Video Corpus Moment Retrieval with Contrastive Learning. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '21), July 11–15, 2021, Virtual Event, Canada*. 11 pages. <https://doi.org/10.1145/3404835.3462874>

1 INTRODUCTION

Video corpus moment retrieval (VCMR) is a domain-specific retrieval task. The aim is to retrieve a short fraction in a video, that semantically corresponds to a text query, from a corpus of untrimmed and unsegmented videos. We use the term *VCMR* to distinguish this task from *single video moment retrieval* (SVMR). As its name suggests, SVMR is to retrieve a short fraction from a single given video that corresponds to a text query. In fact, the task of VCMR was extended from SVMR by Escorcía et al. [16]. VCMR better matches real-world application scenarios, such as query-based video surveillance, search, and navigation, within a video corpus.

Video and text are from two feature spaces. In order to perform query-based video moment retrieval, we need to learn the matching between query and video from training samples. In general, there are two approaches, illustrated in Figure 1. One is to encode video and text separately, and learn the matching through late feature fusion, known as **unimodal encoding** [16, 37]. With unimodal encoding, the text query is encoded to a d -dimensional feature vector. A video is encoded to a sequence of d -dimensional feature vectors, where each vector corresponds to a small fraction of the video, *e.g.*, a few frames. The other is **cross-modal interaction learning**, which takes in a video as a sequence of visual features, and the query as a sequence of word features to learn their interactions [78]. The latter typically leads to better retrieval accuracy as the learned parameters capture the relevance between query and video at fine-grained granularity. However, in query evaluation, cross-modal encoding needs to be performed between query and *every video* in corpus (illustrated by “ $\times N$ ” in Figure 1), leading to high computational cost. On the other hand, with unimodal encoding, visual features can be pre-encoded and stored. In query evaluation, we only need to encode query and then perform video retrieval and moment localization. The challenge becomes to refine two separate encoders during training process, such that the encoded features are well aligned for accurate retrieval. We follow the unimodal encoding approach for its high efficiency. As illustrated in Figure 1,

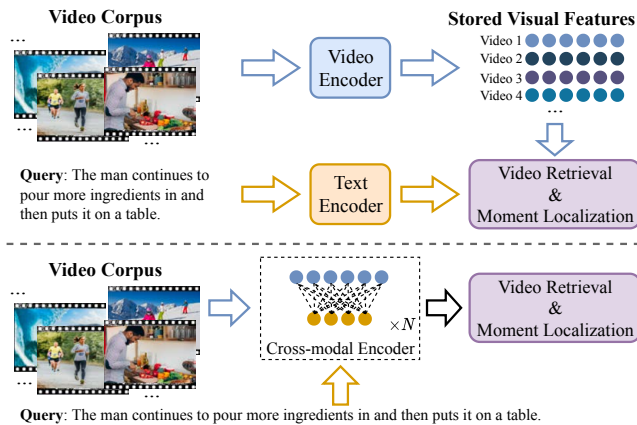


Figure 1: Two approaches to VCMR: unimodal encoding vs. cross-modal interaction learning.

for both approaches, video retrieval and moment localization are performed jointly, *i.e.*, the model is trained with a joint objective. An earlier study [37] has shown that joint learning outperforms two-stage learning where video retrieval and moment localization are treated as two separate subtasks and performed in stages.

The essence of cross-modal interaction is to highlight the relevant and important information from both modalities via co-attention mechanisms. Meanwhile, contrastive learning [22, 25, 71] is a strategy to maximize the mutual information (MI) [5, 33] of positive pairs and to minimize the MI of negative pairs. In our context, a pair of matching video and query is a positive pair and a non-matching pair is a negative pair in training. We consider that both cross-modal interaction learning and contrastive learning share the same objective of emphasizing the relevant information of input pairs. Hence, we can apply contrastive learning to refine encoders in unimodal encoding to achieve similar effectiveness.

In this paper, we develop a Retrieval and Localization Network (ReLoNet) as a base network to separately encode video and query representations, and to (late) fuse them for joint retrieval. We then introduce contrastive learning to ReLoNet to simulate cross-modal interactions between video and query, and propose ReLoCLNet. Build on top of ReLoNet, ReLoCLNet is trained with two contrastive learning objectives: VideoCL and FrameCL. The VideoCL objective aims to learn video and text features such that the semantically related videos and queries are close to each other, and far away otherwise. The FrameCL works at frame-level for moment localization, which simulates fine-grained cross-modal interactions between visual and textual features within a video. In FrameCL, we regard the features within target moment as foreground (positive samples), while the remaining as background (negative samples). Thus, FrameCL enhances the MI between query and foreground, while suppresses the MI between query and background. Once trained, the learned parameters in video encoder and text encoder can be used to encode video and text features separately and independently. Accordingly, all videos in a given corpus can be pre-encoded by the learned video encoder and stored, as illustrated in Figure 1, for efficient retrieval. Our main contributions are as follows:

- To the best of our knowledge, we are the first to address the contradiction between high efficiency and high-quality retrieval in VCMR, by replacing conventional cross-modal interaction learning with contrastive learning.
- We propose two contrastive learning objectives, VideoCL and FrameCL, to simulate cross-modal interactions at both video level and frame level, by measuring the mutual information between video and query at different granularity.
- We conduct experiments on two benchmarks to demonstrate that ReLoCLNet achieves comparable accuracy with cross-modal interaction learning, with much faster retrieval speed. On TVR dataset, ReLoCLNet is about 56 times faster.

2 RELATED WORK

We review related studies on video retrieval, single video moment retrieval, video corpus moment retrieval, and contrastive learning.

Video Retrieval. Given a text query and a set of candidate videos, video retrieval (VR) aims to retrieve and rank candidate videos based on their relevance to the query. Many works [50, 54, 72] jointly model video and text to map them into two holistic representations in a joint embedding space. Their similarities are computed as ranking scores. Venugopalan et al. [64] develop a sequence model for video to text translation and matching. To handle long text query, hierarchical models [58, 79] are proposed to match video and text at different scales. Recently, Li et al. [39] present a text-video matching strategy by using multiple encoders, which can prevent matching from being dominated by a specific encoder.

Single Video Moment Retrieval. SVMR aims to localize a relevant temporal moment in an untrimmed video for a given query [18, 29]. This problem has been well studied and many approaches have been proposed. Ranking-based methods [11, 18, 19, 28, 30, 42, 83] solve SVMR with propose-and-rank pipeline. The given video is pre-segmented into proposals (*i.e.*, video segments) and the proposals are ranked by their similarities to the query. Anchor-based methods [8, 66, 75, 84] replace proposal generation process by assigning each frame with multi-scale anchors sequentially. The anchor (similar to temporal window on video) with highest confidence is selected as result. Regression-based methods [9, 10, 44, 52, 76, 77] directly regress temporal times of target moment through cross-modal interactions learning. Span-based methods [20, 56, 80, 81] follow the concept of extractive question answering (QA) [32, 57, 68, 73]. These methods adopt QA based models to encode multimodal representations for video and query, and predict start and end boundaries of target moment directly. There are also studies [6, 23, 24, 67, 69, 70] formulate SVMR as sequential decision-making problem and design reinforcement learning methods. Other solutions such as weakly supervised learning and jointly training with event captioning have also been explored [10, 41, 51, 58, 65, 69, 82].

Video Corpus Moment Retrieval. Escorcia et al. [16] first extend SVMR to VCMR, and devise a ranking-based clip-query alignment model. The model compares query features with uniformly partitioned video clips. Lei et al. [37] construct a new VCMR dataset named TVR, where the videos come with textual subtitles. The authors propose a proposal-free cross-modal moment localization

(XML) model to jointly learn video retrieval and moment localization objectives. Note that, the “cross-modal” component in XML conceptually is the same as late feature fusion in unimodal encoding approach (see Figure 1). In our classification, the XML model falls under unimodal encoding approach. As a typical cross-modal interaction learning approach, Zhang et al. [78] propose a hierarchical multimodal encoder (HAMMER) to jointly train video retrieval and moment localization with fine-grained cross-modal interactions between query and video. Though effective, HAMMER suffers from low-efficiency and high computational cost. Lastly, Li et al. [38] develops a video-language model for joint representation learning. The model is applied on VCMR for fine-tuning purpose only.

Contrastive learning. Contrastive learning (CL) usually serves as an unsupervised objective to learn representations by contrasting positive pairs against negative pairs [12, 22, 25, 49, 71, 86]. In our context, a positive pair is a matching video-query pair, and a negative pair is a non-matching video and query. One way to achieve contrastive learning is to directly maximize the mutual information (MI) [5, 33] between latent representations [2, 31]. There are also solutions to estimate the lower bounds of MI [4, 53] for unsupervised learning [31, 60, 63]. CL has been applied to vision-language tasks to learn the joint representations of visual and textual modalities [46, 47, 59]. Miech et al. [47] proposes a MIL-NCE objective to address the misalignment between text and video clip in narrated videos for joint representation learning. While MIL-NCE is used for video-text matching, Luo et al. [46] develops a unified pre-training model for multimodal understanding and generation.

3 THE ReLoCLNet MODEL

To ensure retrieval efficiency, we follow the unimodal encoding approach (see Figure 1) and aim to develop video encoder and text encoder for effective feature encoding separately. To achieve high-quality retrieval results, we aim to simulate the cross-modal interaction learning to better align the encoded video and text features. To this end, we introduce contrastive learning to our model. Conceptually, contrastive learning and cross-modal interaction learning share a similar objective of highlighting the relevant information of input pairs, *i.e.*, matching video-query pairs in our setting. Different from cross-modal interaction learning, contrastive learning is only engaged in the training phase. Once trained, the learned parameters ensure the alignment between the encoded video features and text features even though the two features are encoded separately. The task objective, *i.e.*, video retrieval and moment localization, can then be easily achieved through late feature fusion.

In this section, we first develop the ReLoNet as a base model, to separately encode video and query inputs and fuse them for prediction. Then we design two contrastive learning objectives: (i) Video-level Contrastive Learning (VideoCL) for video retrieval, and (ii) Frame-level Contrastive Learning (FrameCL) for moment localization. During training phrase, VideoCL and FrameCL simulate the cross-modal interaction to enhance the representation learning. During inference (*i.e.*, retrieval) phrase, the model separately encodes video and query to maintain retrieval efficiency. The overall architecture of the proposed model is shown in Figure 2. Next, we formally formulate the research problem, then detail the components in ReLoNet and ReLoCLNet.

3.1 Problem Formulation

We denote a video corpus as $\mathcal{V} = \{V^1, V^2, \dots, V^M\}$, where M is the number of videos and $V^k = [f_i]_{i=0}^{T-1}$ represents the k -th video with T frames.² Given a text query $Q = [q_i]_{i=0}^{n_q-1}$, we aim to retrieve the temporal moment (starting from τ^s and ending at τ^e) in V^* that semantically corresponds to Q from video corpus \mathcal{V} . Here V^* denotes a video that contains the ground truth moment and $\tau^{s/e}$ are the start/end time points of target moment in V^* . Thus, VCMR has two objectives: (i) video retrieval, *i.e.*, finding V^* from \mathcal{V} ; and (ii) moment localization, *i.e.*, locating the target moment in V^* .

For words in Q , the initial encoding is obtained from pre-trained word embeddings or language models as $\mathbf{Q} = [q_i]_{i=0}^{n_q-1} \in \mathbb{R}^{d_w \times n_q}$, where d_w is the word feature dimension. For each video $V \in \mathcal{V}$, we split it into n_v clip units, and use pre-trained feature extractor to encode them into visual features $\mathbf{V} = [v_i]_{i=0}^{n_v-1} \in \mathbb{R}^{d_v \times n_v}$, where d_v is the visual feature dimension. Then, $\tau^{s(e)}$ are mapped to the corresponding indices $i^{s(e)}$ in the visual feature sequence, and the target moment is represented as $\mathbf{m}^* = \{v_i | i = i^s, \dots, i^e\}$, where $0 \leq i^s \leq i^e \leq n_v - 1$. That is, in term of visual feature space, \mathbf{m}^* may correspond to a sequence of v_i 's of any length within n_v starting from any index. The best matching \mathbf{m}^* can be estimated by:

$$\mathbf{m}^* = \arg \max_{\mathbf{m} \in \mathcal{V}, \mathbf{V} \in \mathcal{V}} p(\mathbf{m} | \mathbf{V}, \mathbf{Q}) p(\mathbf{V} | \mathbf{Q}) \quad (1)$$

Given M videos in \mathcal{V} with average video feature length n_v , the search space is $\mathcal{O}(M \times n_v^2)$. It is infeasible to compute \mathbf{m}^* in such a large space. Hence, we approximate Eq. 1 by:

$$\mathbf{V}^* = \arg \max_{\mathbf{V}} p(\mathbf{V} | \mathbf{Q}) \quad \text{and} \quad \mathbf{m}^* \approx \arg \max_{\mathbf{m} \in \mathbf{V}^*} p(\mathbf{m} | \mathbf{V}^*, \mathbf{Q}) \quad (2)$$

Eq. 2 is consistent with two objectives of VCMR, *e.g.*, video retrieval and moment localization. The search space reduces to $\mathcal{O}(M + M' \times n_v^2)$, where M' is the top- M' retrieved videos ($M' \ll M$) from the video corpus. In addition to visual features, a video may contain its own multi-modality features, such as subtitle and audio. For instance, videos in TVR dataset [37] come with subtitles. We denote the subtitle of a video by S , and the features extracted from subtitle by $\mathbf{S} \in \mathbb{R}^{d_w \times n_s}$. For easy presentation, we assume all videos come with subtitles and simply use “video” to refer “video + subtitle”.

3.2 Query Encoder

The structure of query encoder is shown in Figure 2. Given a text query Q with n_q words, we first apply textual feature extractor to convert words in the query to corresponding features $\mathbf{Q} = [q_i]_{i=0}^{n_q-1} \in \mathbb{R}^{d_w \times n_q}$. Then we project the obtained features into dimension d with a feed-forward layer as $\hat{\mathbf{Q}} = \mathbf{W}_q \cdot \mathbf{Q} + \mathbf{b}_q \in \mathbb{R}^{d \times n_q}$, where $\mathbf{W}_q \in \mathbb{R}^{d \times d_w}$ and $\mathbf{b}_q \in \mathbb{R}^d$ are the learnable weight and bias, respectively.

Positional embedding is incorporated to each feature of the query sequence $\hat{\mathbf{Q}}$ before they are fed to the transformer blocks [62]. We adopt the transformer block to better capture the contextual representations of the query, for its proven effectiveness [9, 44, 81]. Specifically, the transformer block consists of a multi-head attention layer and a feed-forward layer. Residual connection [26] and layer normalization [1] strategies are applied to each layer in

²Videos in \mathcal{V} could be of different lengths; we simply use T to represent the length (in number of frames) of an arbitrary video.

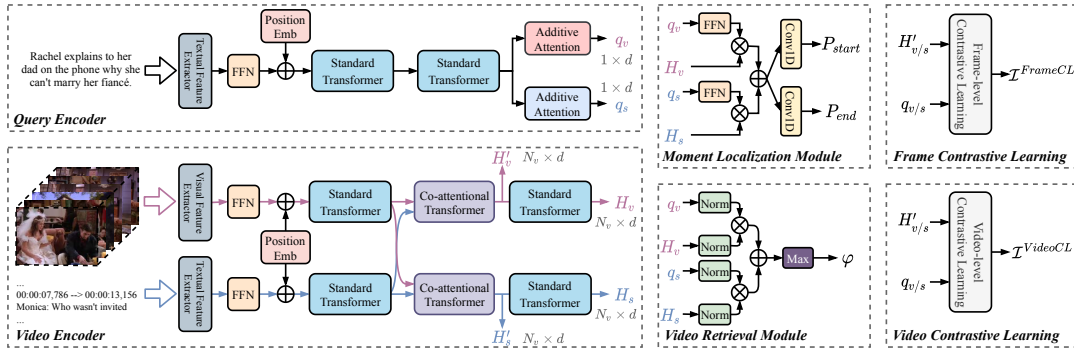


Figure 2: In both ReLoNet and ReLoCLNet, query is encoded to $q_{v/s}$ and video is encoded to $H_{v/s}$, for video retrieval and moment localization. ReLoCLNet adds contrastive learning objectives through $q_{v/s}$ and $H'_{v/s}$ to refine query and video encoders.

the transformer block. The encoded contextual representations of the query after the transformer block become \tilde{Q} .

$$\tilde{Q} = \text{Transformer}_q(\hat{Q}) \quad (3)$$

We use *two* transformer blocks in the query encoder. Then we apply additive attention mechanism [3] to compute the attention scores of each query word. The scores computed are utilized to aggregate the information of $\tilde{Q} = [\tilde{q}_0, \tilde{q}_2, \dots, \tilde{q}_{n_q-1}]$ to compute the modularized query vector, *i.e.*, the sentence representation of \tilde{Q} :

$$\alpha^q = \text{Softmax}(W_{m,\alpha} \cdot \tilde{Q}) \in \mathbb{R}^{n_q}, \quad q_m = \sum_{i=0}^{n_q-1} \alpha_i^q \times q_i \in \mathbb{R}^d \quad (4)$$

where $q_m \in \mathbb{R}^d$ denotes the modularized query vector. $m \in \{v, s\}$ means two modularized query vectors, q_v and q_s , are computed for matching with visual and subtitle features, respectively. Both q_v and q_s are d -dimensional vectors as shown in Figure 2. If the videos to be retrieved do not contain subtitles, then only q_v is computed.

3.3 Video Encoder

We detail the video encoder with the assumption that the videos come with subtitles, as shown in Figure 2. Given a video with its subtitle, we first use visual and textual feature extractors to obtain the corresponding visual and subtitle features $V \in \mathbb{R}^{d_v \times n_v}$ and $S \in \mathbb{R}^{d_w \times n_s}$, respectively. Then both V and S are projected into dimension d with two feed-forward layers as $\hat{V} = W_v \cdot V + b_v \in \mathbb{R}^{d \times n_v}$ and $\hat{S} = W_s \cdot S + b_s \in \mathbb{R}^{d \times n_s}$, where $W_v \in \mathbb{R}^{d \times d_v}$ and $b_v \in \mathbb{R}^d$ are the weight and bias for video feed-forward layer; $W_s \in \mathbb{R}^{d \times d_w}$ and $b_s \in \mathbb{R}^d$ are the weight and bias for subtitle feed-forward layer.

Similar to the query encoder, we add positional embeddings to both \hat{V} and \hat{S} , and feed them to the transformer block. The encoded contextual representations for video and subtitle are:

$$\tilde{V} = \text{Transformer}_v(\hat{V}), \quad \tilde{S} = \text{Transformer}_s(\hat{S}) \quad (5)$$

where $\tilde{V} \in \mathbb{R}^{d \times n_v}$ and $\tilde{S} \in \mathbb{R}^{d \times n_s}$.

Different from the query encoder, we do not use two transformer blocks here. Instead, after the first transformer blocks, we use co-attentional transformer blocks [37, 45, 61, 85]. Because the visual

content in a video and its subtitle are well aligned, through co-attentional transformers, we are able to better capture the cross-modal representations of video and subtitle within a video. Given \tilde{V} and \tilde{S} , the cross-modal representations are encoded as:

$$\begin{aligned} H'_v &= \text{Co-Transformer}_{vs}(\tilde{V}, \tilde{S}) \\ H'_s &= \text{Co-Transformer}_{sv}(\tilde{S}, \tilde{V}) \end{aligned} \quad (6)$$

where $H'_v \in \mathbb{R}^{d \times n_v}$ and $H'_s \in \mathbb{R}^{d \times n_s}$ are the learned cross-modal representations of video and subtitle, respectively.

Finally, we refine the encoded cross-modal representations of H'_v and H'_s with standard transformer blocks by learning the self-attentive contexts, respectively. The final output is calculated as:

$$H_v = \text{Transformer}_v(H'_v), \quad H_s = \text{Transformer}_s(H'_s) \quad (7)$$

where $H_v \in \mathbb{R}^{d \times n_v}$ and $H_s \in \mathbb{R}^{d \times n_s}$ are the final output representations of video and subtitle, respectively.

If videos do not come with subtitles, then the feature encoding pipeline for subtitle will be removed. Accordingly, the co-attentional transformer becomes the standard transformer, and the final output is H_v only.

3.4 Video Retrieval Module

Through query encoding, a query is encoded to two d -dimensional vectors $q_m \in \mathbb{R}^d$, $m \in \{v, s\}$, for matching with visual and subtitle features from a video. Recall that, with video encoding, each video is encoded to $H_m = [h_m^0, h_m^1, \dots, h_m^{n_v-1}] \in \mathbb{R}^{d \times n_v}$, *i.e.*, a sequence of h_m 's each represents two d -dimensional vectors for visual and subtitle features extracted from a small fraction of a video.

We estimate the matching between the query and a video by cosine similarities computed on q_m and H_m , *i.e.*, a simple late feature fusion. Specifically, we compute the cosine similarities between q_m and each element of H_m as:

$$\varphi_m = \text{norm}(H_m^T) \cdot \text{norm}(q_m) \quad (8)$$

where $m \in \{v, s\}$, $\varphi_m \in \mathbb{R}^{n_v}$, and norm denotes the l_2 normalization operation. Then we select the maximum score in φ_m to represent the matching between query and video:

$$\varphi_m = \max(\varphi_m) = \max([\varphi_m^0, \varphi_m^1, \dots, \varphi_m^{n_v-1}]) \quad (9)$$

where φ_m is a scalar. If videos come with subtitles, then $\varphi = \frac{1}{2}(\varphi_v + \varphi_s)$, otherwise $\varphi = \varphi_v$.

We adopt the hinge loss as training objective for video retrieval, similar to [15, 17, 37, 74]. We first sample two sets of negative pairs $\{(Q_i^-, V)\}_{i=1}^N$ and $\{(Q, V_i^-)\}_{i=1}^N$ for each positive pair (Q, V) , where Q^- and V^- denote the negative (*i.e.*, non-matching) query and video, respectively.³ Suppose the computed similarity scores of both sets of negative pairs are φ' and φ'' , the hinge loss is calculated as:

$$\mathcal{L}^{VR} = \max(0, \Delta + \frac{1}{N} \sum \varphi' - \varphi) + \max(0, \Delta + \frac{1}{N} \sum \varphi'' - \varphi) \quad (10)$$

where Δ is the pre-defined margin value and we set $\Delta = 0.1$.

3.5 Moment Localization Module

For efficiency purpose, moment localization is also computed based on the encoded query features \mathbf{q}_m and video features \mathbf{H}_m , through late feature fusion, following [37, 40]. Specifically, \mathbf{q}_m is further encoded with a feed-forward layer as $\mathbf{q}'_m = \mathbf{W}_m \cdot \mathbf{q}_m + \mathbf{b}_m \in \mathbb{R}^d$. Then we compute video-query similarity scores as:

$$\mathbf{S}_{mq} = \mathbf{H}_m^\top \cdot \mathbf{q}'_m \in \mathbb{R}^{n_v}, \quad \text{where } m \in \{v, s\} \quad (11)$$

Again, if subtitle is available, $\mathbf{S} = \frac{1}{2}(\mathbf{S}_{vq} + \mathbf{S}_{sq})$, otherwise $\mathbf{S} = \mathbf{S}_{vq}$. The start and end scores for target moment are generated by convolutional start-end boundary predictor [37]:

$$\mathbf{S}_{\text{start}} = \text{Conv1D}_{\text{start}}(\mathbf{S}), \quad \mathbf{S}_{\text{end}} = \text{Conv1D}_{\text{end}}(\mathbf{S}) \quad (12)$$

where $\mathbf{S}_{\text{start/end}} \in \mathbb{R}^{n_v}$. Then, the probability distributions of start and end boundaries are computed by:

$$\mathbf{P}_{\text{start}} = \text{Softmax}(\mathbf{S}_{\text{start}}), \quad \mathbf{P}_{\text{end}} = \text{Softmax}(\mathbf{S}_{\text{end}}) \quad (13)$$

For a video-query pair, the predicted start and end boundaries of the target moment are derived by maximizing the joint probability:

$$\begin{aligned} (\hat{i}^s, \hat{i}^e) &= \arg \max_{a^s, a^e} \mathbf{P}_{\text{start}}(a^s) \times \mathbf{P}_{\text{end}}(a^e) \\ P^{se} &= \mathbf{P}_{\text{start}}(\hat{i}^s) \times \mathbf{P}_{\text{end}}(\hat{i}^e) \end{aligned} \quad (14)$$

where $0 \leq \hat{i}^s \leq \hat{i}^e \leq n_v - 1$, and P^{se} is the score of best boundaries (\hat{i}^s, \hat{i}^e) . The training objective of moment localization is:

$$\mathcal{L}^{ML} = \frac{1}{2} \times \left(f_{\text{XE}}(\mathbf{P}_{\text{start}}, \mathbf{Y}_{\text{start}}) + f_{\text{XE}}(\mathbf{P}_{\text{end}}, \mathbf{Y}_{\text{end}}) \right) \quad (15)$$

where f_{XE} is the cross-entropy function, $\mathbf{Y}_{\text{start}}$ and \mathbf{Y}_{end} are one-hot labels for start (i^s) and end (i^e) boundaries of the ground truth moment, respectively.

We now have the full picture of the base architecture ReLoNet with four modules: query encoder (Section 3.2), video encoder (Section 3.3), video retrieval and moment localization modules (Sections 3.4 and 3.5). Next, we incorporate contrastive learning objectives into ReLoNet to develop ReLoCLNet.

3.6 Video and Frame Contrastive Learning

In ReLoNet, video retrieval and moment localization are fully based on the encoded query features \mathbf{q}_m and video features \mathbf{H}_m . They are both computed by simple late future fusion. Quality of the final moment retrieval hence heavily relies on the effectiveness of the two separate encoders, query encoder and video encoder.

³We simply use V to represent a video with its subtitle if available.

In ReLoCLNet, we aim to guide the two encoders to simulate cross-modal interaction learning in the training phase. To this end, we introduce two contrastive learning objectives, VideoCL and FrameCL. VideoCL guides the two encoders to better distinguish matching video-query pairs from non-matching pairs. FrameCL guides the two encoders to better distinguish the matching moment to the query from the non-matching moments.

3.6.1 Video Contrastive Learning (VideoCL). VideoCL guides the encoders to learn a joint feature space where the semantically related videos and queries are close to each other, and far away otherwise. In other words, VideoCL aims to reduce the distance of matching video-query pairs, and to increase the distance of non-matching pairs, in the joint feature space.

We encode the latent representation of video $\mathbf{H}'_m \in \mathbb{R}^{d \times n_v}$ from Eq. 6 (illustrated as \mathbf{H}'_v and \mathbf{H}'_s in Figure 2) into its modularized video representation \mathbf{c}_m . Similar to modular component in query encoder, we adopt additive attention mechanism to compute \mathbf{c}_m :

$$\alpha^m = \text{Softmax}(\mathbf{W}_{m,\alpha} \cdot \mathbf{H}'_m) \in \mathbb{R}^{n_v}, \quad \mathbf{c}_m = \sum_{i=0}^{n_v-1} \alpha_i^m \times \mathbf{h}'_{m,i} \quad (16)$$

where $\mathbf{c}_m \in \mathbb{R}^d$, $\mathbf{W}_{m,\alpha} \in \mathbb{R}^{1 \times d}$ and $m \in \{v, s\}$.

Given a set of positive (*i.e.*, matching) video-query pairs $\mathcal{P} = \{(\mathbf{c}_m, \mathbf{q}_m)\}$ and the sampled set of negative (*i.e.*, non-matching) video-query pairs $\mathcal{N} = \{(\mathbf{c}'_m, \mathbf{q}'_m)\}$, we adopt the noise-contrastive estimation (NCE) [21, 34, 47, 59] to compute the VideoCL score:

$$\mathcal{I}_m^e = \log \left(\frac{\sum_{(\mathbf{c}_m, \mathbf{q}_m) \in \mathcal{P}} e^{f(\mathbf{c}_m)^\top \cdot g(\mathbf{q}_m)}}{\sum_{(\mathbf{c}_m, \mathbf{q}_m) \in \mathcal{P}} e^{f(\mathbf{c}_m)^\top \cdot g(\mathbf{q}_m)} + \sum_{(\mathbf{c}'_m, \mathbf{q}'_m) \sim \mathcal{N}} e^{f(\mathbf{c}'_m)^\top \cdot g(\mathbf{q}'_m)}} \right) \quad (17)$$

where the exponential term, $e^{f(\mathbf{c})^\top \cdot g(\mathbf{q})}$, computes the mutual information (MI) between \mathbf{c} and \mathbf{q} . $f(\cdot)$ and $g(\cdot)$ denote the parametrized mappings, which project video and query representations into the same embedding space. Again, $\mathcal{I}^e = \frac{1}{2}(\mathcal{I}_v^e + \mathcal{I}_s^e)$ if subtitle is available, otherwise $\mathcal{I}^e = \mathcal{I}_v^e$.

The objective of NCE is to optimize $\max_{f,g}(\mathcal{I}^e)$, which is equivalent to maximizing the ratio of the summed MI's of all samples in \mathcal{P} and the summed MI's of all samples in \mathcal{N} [47]. The loss of VideoCL is defined as:

$$\mathcal{L}^{\text{VideoCL}} = -\mathcal{I}^e \quad (18)$$

3.6.2 Frame Contrastive Learning (FrameCL). FrameCL focuses on moment localization within a given pair of video-query, where the video retrieval module predicts the video contains a matching moment to the query. We regard the video features that reside within boundaries of the target moment as foreground or positive samples, and the rest as background or negative samples. Then we compute the contrastive loss by measuring MI between the query and the positive/negative video features. For this purpose, we utilize a discriminative approach based on mutual information maximization [31, 63].

The structure of FrameCL module is shown in Figure 3. The inputs \mathbf{H}'_v , \mathbf{H}'_s , \mathbf{q}_v , and \mathbf{q}_s are outputs illustrated in Figure 2. Given the latent representation of video $\mathbf{H}'_m \in \mathbb{R}^{d \times n_v}$, we first split it into two parts by boundaries of target moment. The positive/foreground

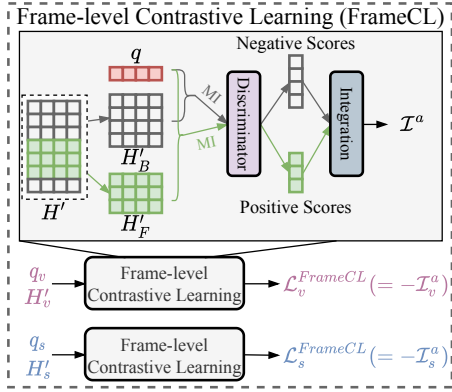


Figure 3: Structure of the FrameCL module.

video features are $H'_{m,F} = \{h'_{m,i} | i = i^s, \dots, i^e\} \in \mathbb{R}^{d \times n_t}$, which are features within the target moment.⁴ The negative/background features $H'_{m,B} = \{h'_{m,i} | i = 0, \dots, i^s - 1, i^e + 1, \dots, n_v - 1\} \in \mathbb{R}^{d \times (n_v - n_t)}$, are not in the target moment.

With query representation q_m , foreground representation $H'_{m,F}$, and background representation $H'_{m,B}$, our goals are to maximize the MI between the query and the foreground, as well as to minimize the MI between the query and the background. Since MI estimation is in general intractable for continuous and random variables, we choose to maximize the value over lower bound estimators of MI, through Jensen-Shannon MI estimator [31] as:

$$\mathcal{I}_m^a = \mathbb{E}_{H'_{m,F}} \left[-\text{sp}(-C_\theta(q, H'_{m,F})) \right] - \mathbb{E}_{H'_{m,B}} \left[\text{sp}(C_\theta(q, H'_{m,B})) \right] \quad (19)$$

where $\text{sp}(x) = \log(1 + ex)$ is the Softplus activation. $C_\theta : d \times d \rightarrow \mathbb{R}$ refers to a discriminator. Similarly, $\mathcal{I}^a = \frac{1}{2}(\mathcal{I}_v^a + \mathcal{I}_s^a)$ if subtitle is available, otherwise $\mathcal{I}^a = \mathcal{I}_v^a$. The contrastive loss of FrameCL is:

$$\mathcal{L}^{\text{FrameCL}} = -\mathcal{I}^a \quad (20)$$

Note that, both VideoCL and FrameCL are training objectives, and their losses are used to update video and query encoders. Although the two objectives are designed for video retrieval and moment localization respectively, they mutually affect each other, because both video and query encoders are adjusted based on the loss from both VideoCL and FrameCL, together with other losses.

3.7 Training and Inference

The overall training loss for ReLoCLNet is:

$$\mathcal{L} = \lambda_1 \times \mathcal{L}^{\text{VR}} + \lambda_2 \times \mathcal{L}^{\text{ML}} + \lambda_3 \times \mathcal{L}^{\text{VideoCL}} + \lambda_4 \times \mathcal{L}^{\text{FrameCL}} \quad (21)$$

λ_i 's are hyperparameters to balance the contribution of each loss. We set $\lambda_1 = 1.0$ and $\lambda_{2,3,4} = 0.01$ to keep all losses at the same order of magnitude *i.e.*, equal contributions from the four components. Note that each video contains a large number of candidate moments.

During inference for VCMR, given a text query and a video corpus with M videos, we first use Eq. 8 and 9 to compute the similarity between the query and each of the M videos, leading to $\varphi = [\varphi_1, \varphi_2, \dots, \varphi_M]$. The top- K most relevant videos are retrieved

⁴ $n_t = i^e - i^s + 1$, and it denotes the length of target moment.

Table 1: The hyper-parameters for TVR and ANetCaps

Hyperparameter Name	TVR	ANetCaps
n_v (max video sequence)	128	
n_q (max query sequence)	30	64
d_v (visual feature dim)	3072 (2048(ResNet)+1024(I3D))	1024 (I3D)
d_w (word feature dim)	768 (RoBERTa)	300 (GloVe)
d (hidden size)	384	
γ	30	20
# negative samples in VR: 10	Optimizer: AdamW [14]	
Dropout rate: 0.1	Weight decay rate: 0.01	Batch size: 128
Learning rate (lr): 0.0001	lr warmup proportion: 0.01	
Early stop tolerance: 10	# total training epochs: 100	

based on φ ($K = 100$ in our implementation). For each retrieved video, we compute the scores of a few candidate predicted moments by Eq. 14. Let P^{se} be the score of one predicted moment in the video. The final VCMR score is computed by:

$$\delta = P^{se} \times e^{\gamma \cdot \varphi} \quad (22)$$

The exponential term and the hyperparameter γ are used to balance the importance of video retrieval and moment localization scores.

4 EXPERIMENTS

4.1 Datasets and Evaluation Metrics

We conduct experiments on two benchmark datasets: ActivityNet Captions [36] and TVR [37]. ActivityNet Captions (**ANetCaps**) contains around 20K videos taken from the ActivityNet [27] dataset. The average video duration is about 120 seconds, the average query length is around 14.78 words, the average moment duration is about 36.18 seconds, and each video contains 3.68 annotations on average. This dataset is originally designed for SVMR task, then adapted to VCMR by Escorcía et al. [16]. We follow the setup in [16, 78] with 10, 009 and 4, 917 videos (*i.e.*, 37, 421 and 17, 505 annotations) for train and test, respectively. **TVR** is collected by Lei et al. [37], which contains 21.8K videos and 109K queries in total. The average video duration is 76.2 seconds, the query contains 13.4 words on average, the average moment duration is 9.1 seconds, and each video contains 5 annotations on average. We follow Zhang et al. [78] with 17, 435 and 2, 179 videos for train and test, respectively. Same as Lei et al. [37] and Zhang et al. [78], we utilize both video and subtitle features in the TVR dataset for train and test.

We evaluate the models for the VCMR task as well as its two sub-tasks: video retrieval (VR) and SVMR. For VR, we use “**Recall@ k** ” ($k \in \{1, 5, 10, 100\}$) as the evaluation metric following [37, 78]. Note that we do not use “**Precision@ k** ” because each query only corresponds to one ground truth video, in both datasets. For SVMR and VCMR, we use “**Recall@ k , IoU= μ** ” as the evaluation metric, which denotes the percentage of test samples that have at least one predicted moment whose *intersection over union* (IoU) with the ground-truth moment is larger than μ in the top- k predictions. We set $k \in \{1, 10, 100\}$ and $\mu \in \{0.5, 0.7\}$. A prediction is correct if (i) the predicted video matches the ground truth video, and (ii) the predicted moment has high overlap with the ground truth moment, where temporal IoU is used to measure the overlap [37].

Table 2: Results of VCMR on TVR and ANetCaps datasets

Dataset	Method	Recall@ k , IoU = 0.5			Recall@ k , IoU = 0.7		
		R1	R10	R100	R1	R10	R100
TVR	XML [37]	-	-	-	2.62	9.05	22.47
	HERO [38]	-	-	-	2.98	10.65	18.25
	FLAT [78]	8.45	21.14	30.75	4.61	11.29	16.24
	HAMMER [78]	9.19	<i>21.28</i>	31.25	5.13	<i>11.38</i>	16.71
	ReLoNet	5.46	16.65	35.08	2.71	9.37	22.87
	ReLoCLNet	8.03	21.37	44.10	4.15	14.06	32.42
ANetCaps	MCN [30]	0.02	0.18	1.26	0.01	0.09	0.70
	CAL [16]	0.21	1.32	6.82	0.12	0.89	4.79
	FLAT [78]	2.57	<i>13.07</i>	<i>30.66</i>	1.51	7.69	17.67
	HAMMER [78]	2.94	14.49	32.49	1.74	8.75	19.08
	ReLoNet	2.16	9.96	24.54	1.26	5.64	17.43
	ReLoCLNet	3.09	11.28	25.95	1.82	6.91	<i>18.33</i>

Table 3: Retrieval efficiency on the TVR dataset

Method	Retrieval Efficiency	
	Total Time	Average Per Query
XML [37]	39.34 seconds	3.61 milliseconds
HAMMER [78]	2,378.67 seconds	218.33 milliseconds
ReLoNet	42.07 seconds	3.86 milliseconds
ReLoCLNet		

4.2 Implementation Details

For ANetCaps, we use I3D [7] pre-trained on Kinetics dataset [35] as the visual feature extractor following Zhang et al. [78], and adopt GloVe embeddings [55] as the textual feature extractor for query words. For TVR, we directly use the visual and textual features provided by Lei et al. [37]. The visual feature is the concatenation of appearance feature extracted by ResNet152 [26] pre-trained on ImageNet [13] and temporal feature extracted by I3D. The textual feature of query and subtitle is extracted by 12-layer pre-trained RoBERTa [43]. The negative sets of video retrieval and VideoCL modules are sampled within each mini-batch during training. The hyperparameters are summarized in Table 1. Other hyperparameters are given when describing the corresponding model components. Our model is implemented in PyTorch 1.7.0 with CUDA 11.1 and cudnn 8.0.5. All experiments are conducted on a workstation with dual NVIDIA GeForce RTX 3090 GPUs.

4.3 Performance Comparison

We compare our models with MCN [30], CAL [16], XML [37], HERO [38], FLAT [78] and HAMMER [78]. Among them, MCN, CAL, XML, and HERO follow unimodal encoding approaches, while FLAT and HAMMER belong to cross-modal interaction learning approaches (see Figure 1). FLAT is a variant of HAMMER without using hierarchical structure. In all tables, results of the compared models are reported in their corresponding papers.⁵ The best results are in **boldface** and the second bests are in *italic*.

⁵Two sets of results are reported for HERO in [38], with and without large-scale pre-training. We choose the version without pre-training as all other models compared here do not use pre-training.

Table 4: Results of VR subtask on TVR and ANetCaps datasets

Dataset	Method	Recall@ k			
		$k = 1$	$k = 5$	$k = 10$	$k = 100$
TVR	MCN [30]	0.05	0.38	0.66	3.59
	CAL [16]	0.28	1.02	1.68	8.55
	MEE [48]	7.56	20.78	29.88	73.07
	XML [37]	16.54	38.11	50.41	88.22
	ReLoNet	<i>16.96</i>	<i>39.28</i>	<i>51.34</i>	<i>88.46</i>
	ReLoCLNet	22.13	45.85	57.25	90.21
ANetCaps	FLAT [78]	5.37	-	29.14	71.64
	HAMMER [78]	5.89	-	30.98	73.38
	ReLoNet	<i>7.02</i>	<i>24.42</i>	<i>35.24</i>	<i>78.08</i>
	ReLoCLNet	9.64	28.02	40.26	79.13

Table 5: Results of SVMR subtask on TVR and ANetCaps datasets

Dataset	Method	Recall@1, IoU = μ		
		$\mu = 0.3$	$\mu = 0.5$	$\mu = 0.7$
TVR	MCN [30]	-	13.08	5.06
	CAL [16]	-	12.07	4.68
	ExCL [20]	-	<i>31.34</i>	<i>14.19</i>
	XML [37]	-	30.75	13.41
	ReLoNet	<i>48.14</i>	29.49	13.13
	ReLoCLNet	49.87	31.88	15.04
ANetCaps	FLAT [78]	<i>57.58</i>	<i>39.60</i>	<i>22.59</i>
	HAMMER [78]	59.18	41.45	24.27
	ReLoNet	39.27	23.67	14.55
	ReLoCLNet	42.65	28.54	17.76

The results of VCMR on TVR and ANetCaps datasets are reported in Table 2. On TVR dataset, ReLoNet is comparable to XML with slightly better performance. ReLoCLNet outperforms all baselines over Recall@10 and Recall@100 metrics. Observe that the performance of ReLoCLNet is lower than FLAT and HAMMER over Recall@1. Since both FLAT and HAMMER adopt fine-grained cross-modal interaction learning, they are more adequate to align video and query for precise moment retrieval. Compared with ReLoNet, ReLoCLNet achieves significant improvements over all evaluation metrics, which demonstrate the effectiveness of the proposed contrastive learning objectives.

On ANetCaps dataset, ReLoNet surpasses the ranking-based methods, MCN and CAL, by large margins over all evaluation metrics. Similarly, ReLoCLNet is superior to ReLoNet thanks to the contrastive learning components. Compared with FLAT and HAMMER, ReLoCLNet outperforms both over Recall@1 but is poorer over Recall@10 and Recall@100. This observation is contrary to that on TVR dataset. Recall that FLAT and HAMMER adopt cross-modal interactions learning between video and query, and we have separate encoders for video and query. In addition, FLAT and HAMMER utilize pre-trained RoBERTa to extract textual features for query, while we simply adopt GloVe embeddings. All these contribute the differences between our results and that of FLAT and HAMMER. Overall, we consider ReLoCLNet achieves comparable effectiveness with FLAT and HAMMER.

Table 6: The effects of different objectives on TVR dataset (VR=Video Retrieval, ML=Moment Localization, VideoCL=Video Contrastive Learning, and FrameCL=Frame Contrastive Learning)

Objective				VCMR						VR			SVMR					
				Recall@k, IoU=0.5			Recall@k, IoU=0.7			Recall@k			Recall@k, IoU=0.5			Recall@k, IoU=0.7		
VR	ML	VideoCL	FrameCL	1	10	100	1	10	100	1	10	100	1	10	100	1	10	100
✓	✗	✗	✗	-	-	-	-	-	-	16.23	49.33	87.38	-	-	-	-	-	-
✗	✓	✗	✗	-	-	-	-	-	-	-	-	-	30.21	59.81	83.43	13.91	41.55	68.51
✗	✗	✓	✗	5.46	16.65	35.08	2.71	9.37	22.87	16.96	51.34	88.46	29.49	54.06	75.89	13.13	35.46	58.84
✓	✓	✓	✗	6.63	18.16	39.69	3.24	11.78	27.69	20.69	55.70	89.71	29.52	57.32	78.65	13.76	38.26	64.27
✓	✓	✗	✓	7.21	20.04	42.45	3.75	12.77	30.32	19.81	54.38	88.96	31.75	62.20	85.99	14.73	44.60	71.44
✓	✓	✓	✓	8.03	21.37	44.10	4.15	14.06	32.42	22.13	57.25	90.21	31.88	63.89	86.67	15.04	45.24	72.12

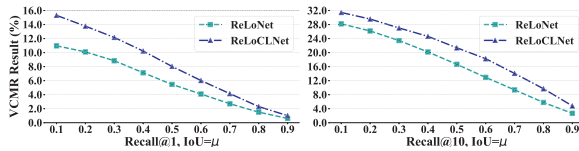


Figure 4: Recall@1 and Recall@10 of VCMR on TVR dataset over different IoU thresholds.

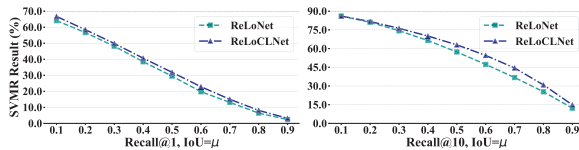


Figure 5: Recall@1 and Recall@10 of SVMR on TVR dataset over different IoU thresholds.

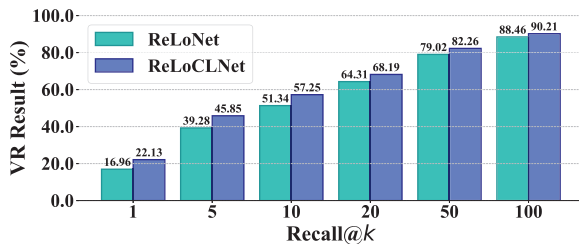


Figure 6: Recall@K of VR on TVR dataset over different K.

4.4 Retrieval Efficiency and Ablation Study

In this section, we compare retrieval efficiency and perform in-depth ablation studies. We study the performance of our models on VR and SVMR subtasks, and the effects of different components.

4.4.1 Retrieval Efficiency. We consider VCMR in the validation set of TVR dataset containing 2, 179 videos with 10, 895 queries. The retrieval efficiency is summarized in Table 3. The time spent on data pre-processing and feature extraction by pre-trained extractor are not counted since the same process applies to all methods. We used the XML code released by the authors, and re-implemented HAMMER according to their paper as its code is not released. Observe that the retrieval efficiency of our models are comparable

to XML, and our models are far more efficient than HAMMER. Although HAMMER performs better on more strict metrics (e.g., Recall@1, IoU=0.7), our models are around 56.71 times faster than HAMMER in retrieval. Note that, ReLoCLNet and ReLoNet have the same retrieval efficiency, because neither VideoCL nor FrameCL introduces additional parameters; and all additional computations of ReLoCLNet happen in training stage.

4.4.2 Video Retrieval Subtask. Table 4 reports the results on TVR and ANetCaps datasets. Observe that ReLoNet performs slightly better than XML on TVR, and significantly better than HAMMER on ANetCaps. ReLoCLNet outperforms all baselines by large margins on both datasets. In particular, ReLoCLNet achieves 5.59% improvement in Recall@1 comparing with XML on TVR dataset. On ANetCaps dataset, ReLoCLNet obtains 9.64% absolute score in Recall@1, compared with 5.89% of HAMMER.

4.4.3 Single Video Moment Retrieval Subtask. The results of SVMR on both datasets are reported in Table 5. On TVR, ReLoCLNet achieves best performance, and obtains significant improvements against ReLoNet. Compared with ExCL, ReLoCLNet only outperforms by a small margin. ExCL is specially designed for SVMR, with fine-grained cross-modal interactions learning. On ANetCaps, ReLoCLNet is superior to ReLoNet by large margins, which again shows the effectiveness of contrastive learning. However, ReLoCLNet performs worse than FLAT and HAMMER. Because both FLAT and HAMMER inherit their architectures designed for SVMR, which contain sophisticated and computational expensive cross-modal interactions for high-quality moment retrieval. In contrast, ReLoCLNet only relies on simple late fusion of separately encoded query and video features.

4.4.4 Analysis on the Learning Objectives. Table 6 reports the contributions of different training objectives on TVR dataset. Note ReLoNet equals to VR+ML objectives, and ReLoCLNet is with all the four objectives. We first analyze the video retrieval (VR) and moment localization (ML) objectives. ReLoNet jointly trains VR and ML objectives for the VCMR task. Comparing VR with ReLoNet, the performance of ReLoNet on video retrieval is slightly better than that of VR, which means the ML objective also contributes to refine video retrieval learning process. In contrast, compared to ML only, ReLoNet underperforms ML on moment localization with marginal performance degradation, which implies that VR objective has negligible negative impact on moment localization.

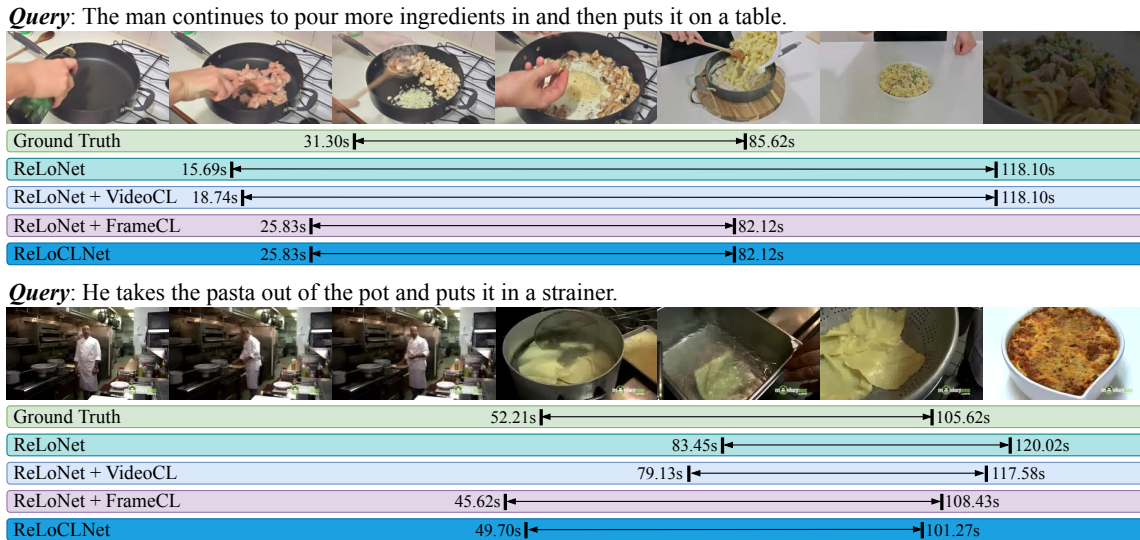


Figure 7: Visualization of moment localization predictions by ReLoNet, ReLoCLNet, and ReLoNet with VideoCL or FrameCL, for two queries on ANetCaps dataset.

Now we analyze the effects of VideoCL and FrameCL objectives. Observe that VideoCL contributes to performance improvements on both VCMR and VR, while it achieves marginal improvements on SVMR. Recall that VideoCL adopts noise-contrastive estimation to enlarge the similarities of matched video-query pairs, and reduce similarities between unpaired videos and queries; this is in line with video retrieval objective. Thus, it is beneficial to video retrieval learning. ReLoNet with FrameCL outperforms ReLoNet on all the three tasks. FrameCL aims to distinguish the matching moment from non-matching moment within a video. In this case, FrameCL guides the model to search for boundaries of target moment for precise moment localization. In fact, the matching between query and video is largely based on the matching moment in the video. In this sense, by highlighting matching moment, FrameCL does contribute to video retrieval task as well. Combining VideoCL and FrameCL, ReLoCLNet further boosts the performances on all three tasks by incorporating the advantages of both VideoCL and FrameCL.

4.5 Qualitative Analysis

Figure 4 plots Recall@1 and Recall@10 of VCMR performances on TVR dataset over different IoU thresholds. We evaluate 9 different IoU(μ) values, from 0.1 to 0.9. ReLoCLNet consistently outperforms ReLoNet, and relative performance improvements of ReLoCLNet are larger under more strict metrics. For instance, compared with ReLoNet, ReLoCLNet achieves 47.07% relative gains (8.03 vs 5.46) in Recall@1, IoU=0.5 versus 28.35% relative gains (21.37 vs 16.65) in Recall@10, IoU=0.5.

Figure 5 plots Recall@1 and Recall@10 of SVMR over different IoU thresholds, and similar observations hold on this task. Figure 6 plots the video retrieval (VR) results of ReLoNet and ReLoCLNet over different recall thresholds on TVR dataset. Similarly, ReLoCLNet surpasses ReLoNet over all thresholds, and the relative performance improvement ratio is larger under more strict metrics.

Finally, we show two retrieval examples in Figure 7 from ANetCaps dataset. The figure shows the predicted moments by ReLoCLNet and ReLoNet+FrameCL are closer to ground truth than that by ReLoNet and ReLoNet+VideoCL, which demonstrates the effectiveness of FrameCL module. Note FrameCL is designed to maximize the mutual information between query and frames within the target moment, and to minimize the MI between the query and frames outside target moment. With FrameCL, the model is guided to search for the boundaries within the region of target moment.

5 CONCLUSION

In this paper, we propose a Retrieval and Localization Network with Contrastive Learning (ReLoCLNet) for video corpus moment retrieval (VCMR) task. Specifically, we introduce two contrastive learning objectives (VideoCL and FrameCL) on top of a unimodal encoding approach, ReLoNet, to address the contradiction between retrieval efficiency and retrieval quality. The VideoCL objective guides the video and query encoders to shorten the distance of matching videos and queries while enlarge the non-matching pairs. The FrameCL objective works at frame-level to simulate the fine-grained cross-modal interactions between visual and textual features within a video. Through extensive experimental studies, we show that ReLoCLNet addresses VCMR with high efficiency, and its retrieval accuracy is comparable with state-of-the-art methods which are much costly in terms of computation. Compared with the expensive cross-model interaction learning, we show that unimodal encoding with contrastive learning is a promising direction to explore for video corpus moment retrieval.

ACKNOWLEDGMENTS

This research is supported by the Agency for Science, Technology and Research (A*STAR) under its AME Programmatic Fund (Project No. A18A1b0045 and A18A2b0046).

REFERENCES

- [1] Jimmy Ba, J. Kiros, and Geoffrey E. Hinton. 2016. Layer Normalization. *ArXiv abs/1607.06450* (2016).
- [2] Philip Bachman, R Devon Hjelm, and William Buchwalter. 2019. Learning Representations by Maximizing Mutual Information Across Views. In *Advances in Neural Information Processing Systems*, Vol. 32. Curran Associates, Inc., 15535–15545.
- [3] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural Machine Translation by Jointly Learning to Align and Translate. In *International Conference on Learning Representations*.
- [4] Mohamed Ishmael Belghazi, Aristide Baratin, Sai Rajeswar, Sherjil Ozair, Yoshua Bengio, Aaron Courville, and R Devon Hjelm. 2018. Mine: mutual information neural estimation. *arXiv preprint arXiv:1801.04062* (2018).
- [5] Anthony J Bell and Terrence J Sejnowski. 1995. An information-maximization approach to blind separation and blind deconvolution. *Neural computation* 7, 6 (1995), 1129–1159.
- [6] Da Cao, Yawen Zeng, Meng Liu, Xiangnan He, Meng Wang, and Zheng Qin. 2020. STRONG: Spatio-Temporal Reinforcement Learning for Cross-Modal Video Moment Localization. In *Proceedings of the 28th ACM International Conference on Multimedia* (Seattle, WA, USA). 4162–4170.
- [7] João Carreira and Andrew Zisserman. 2017. Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset. In *IEEE Conference on Computer Vision and Pattern Recognition*. 4724–4733.
- [8] Jingyuan Chen, Xinpeng Chen, Lin Ma, Zequn Jie, and Tat-Seng Chua. 2018. Temporally Grounding Natural Sentence in Video. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. 162–171.
- [9] Long Chen, Chujie Lu, Siliang Tang, Jun Xiao, Dong Zhang, Chilie Tan, and Xiaolin Li. 2020. Rethinking the Bottom-Up Framework for Query-based Video Localization. In *Proceedings of the AAAI Conference on Artificial Intelligence*.
- [10] Shaoxiang Chen, Wenhao Jiang, Wei Liu, and Yu-Gang Jiang. 2020. Learning Modality Interaction for Temporal Sentence Localization and Event Captioning in Videos. In *The European Conference on Computer Vision*.
- [11] Shaoxiang Chen and Yu-Gang Jiang. 2019. Semantic proposal for activity localization in videos via sentence query. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33. 8199–8206.
- [12] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. 2020. A Simple Framework for Contrastive Learning of Visual Representations. In *Proceedings of the 37th International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 119)*. PMLR, 1597–1607.
- [13] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*. Ieee, 248–255.
- [14] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics, 4171–4186.
- [15] Jianfeng Dong, Xirong Li, Chaoxi Xu, Shouling Ji, Yuan He, Gang Yang, and Xun Wang. 2019. Dual encoding for zero-example video retrieval. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 9346–9355.
- [16] Victor Escorcia, Mattia Soldan, Josef Sivic, Bernard Ghanem, and Bryan Russell. 2019. Temporal Localization of Moments in Video Collections with Natural Language. *arXiv preprint arXiv:1907.12763* (2019).
- [17] Valentin Gabeur, Chen Sun, Karteek Alahari, and Cordelia Schmid. 2020. Multi-modal transformer for video retrieval. In *European Conference on Computer Vision (ECCV)*, Vol. 5. Springer.
- [18] Jiyang Gao, Chen Sun, Zhenheng Yang, and Ramakant Nevatia. 2017. TALL: Temporal Activity Localization via Language Query. In *IEEE International Conference on Computer Vision*. 5277–5285.
- [19] Runzhou Ge, Jiyang Gao, Kan Chen, and Ram Nevatia. 2019. MAC: Mining Activity Concepts for Language-based Temporal Localization. In *IEEE Winter Conference on Applications of Computer Vision*. 245–253.
- [20] Soham Ghosh, Anuva Agarwal, Zarana Parekh, and Alexander Hauptmann. 2019. ExCL: Extractive Clip Localization Using Natural Language Descriptions. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. 1984–1990.
- [21] Michael Gutmann and Aapo Hyvärinen. 2010. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics (Proceedings of Machine Learning Research, Vol. 9)*. JMLR Workshop and Conference Proceedings, Chia Laguna Resort, Sardinia, Italy, 297–304.
- [22] Raia Hadsell, Sumit Chopra, and Yann LeCun. 2006. Dimensionality reduction by learning an invariant mapping. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, Vol. 2. IEEE, 1735–1742.
- [23] Meera Hahn, Asim Kadav, James M Rehg, and Hans Peter Graf. 2020. Tripping through time: Efficient localization of activities in videos. In *The British Machine Vision Conference*.
- [24] Dongliang He, Xiang Zhao, Jizhou Huang, Fu Li, Xiao Liu, and Shilei Wen. 2019. Read, watch, and move: Reinforcement learning for temporally grounding natural language descriptions in videos. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33. 8393–8400.
- [25] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. 2020. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 9729–9738.
- [26] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 770–778.
- [27] F. C. Heilbron, V. Escorcia, B. Ghanem, and J. C. Niebles. 2015. ActivityNet: A large-scale video benchmark for human activity understanding. In *IEEE Conference on Computer Vision and Pattern Recognition*. 961–970.
- [28] Lisa Anne Hendricks, Oliver Wang, Eli Shechtman, Josef Sivic, Trevor Darrell, and Bryan Russell. 2018. Localizing Moments in Video with Temporal Language. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. 1380–1390.
- [29] Lisa Anne Hendricks, Oliver Wang, Eli Shechtman, Josef Sivic, Trevor Darrell, and Bryan C. Russell. 2017. Localizing Moments in Video with Natural Language. In *2017 IEEE International Conference on Computer Vision (ICCV)*. 5804–5813.
- [30] Lisa Anne Hendricks, Oliver Wang, Eli Shechtman, Josef Sivic, Trevor Darrell, and Bryan C. Russell. 2017. Localizing Moments in Video with Natural Language. In *2017 IEEE International Conference on Computer Vision*. 5804–5813.
- [31] R Devon Hjelm, Alex Fedorov, Samuel Lavoie-Marchildon, Karan Grewal, Phil Bachman, Adam Trischler, and Yoshua Bengio. 2019. Learning deep representations by mutual information estimation and maximization. In *International Conference on Learning Representations*.
- [32] Hsin-Yuan Huang, Chenguang Zhu, Yelong Shen, and Weizhu Chen. 2018. FusionNet: Fusing via Fully-aware Attention with Application to Machine Comprehension. In *International Conference on Learning Representations*.
- [33] Aapo Hyvärinen and Erkki Oja. 2000. Independent component analysis: algorithms and applications. *Neural networks* 13, 4-5 (2000), 411–430.
- [34] Rafal Jozefowicz, Oriol Vinyals, Mike Schuster, Noam Shazeer, and Yonghui Wu. 2016. Exploring the limits of language modeling. *arXiv preprint arXiv:1602.02410* (2016).
- [35] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. 2017. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950* (2017).
- [36] R. Krishna, K. Hata, F. Ren, L. Fei-Fei, and J. C. Niebles. 2017. Dense-Captioning Events in Videos. In *IEEE International Conference on Computer Vision*. 706–715.
- [37] Jie Lei, Licheng Yu, Tamara L Berg, and Mohit Bansal. 2020. TVR: A Large-Scale Dataset for Video-Subtitle Moment Retrieval. In *The European Conference on Computer Vision*.
- [38] Linjie Li, Yen-Chun Chen, Yu Cheng, Zhe Gan, Licheng Yu, and Jingjing Liu. 2020. HERO: Hierarchical Encoder for Video+Language Omni-representation Pre-training. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics, Online, 2046–2065.
- [39] Xirong Li, Fangming Zhou, Chaoxi Xu, Jiaqi Ji, and Gang Yang. 2020. SEA: Sentence Encoder Assembly for Video Retrieval by Textual Queries. *IEEE Transactions on Multimedia* (2020).
- [40] Tianwei Lin, Xu Zhao, Haisheng Su, Chongjing Wang, and Ming Yang. 2018. Bsn: Boundary sensitive network for temporal action proposal generation. In *Proceedings of the European Conference on Computer Vision*. 3–19.
- [41] Zhijie Lin, Zhou Zhao, Zhu Zhang, Qi Wang, and Huasheng Liu. 2020. Weakly-Supervised Video Moment Retrieval via Semantic Completion Network. In *Proceedings of the AAAI Conference on Artificial Intelligence*.
- [42] Meng Liu, Xiang Wang, Liqiang Nie, Qi Tian, Baoquan Chen, and Tat-Seng Chua. 2018. Cross-modal Moment Localization in Videos. In *Proceedings of the 26th ACM International Conference on Multimedia*. 843–851.
- [43] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692* (2019).
- [44] Chujie Lu, Long Chen, Chilie Tan, Xiaolin Li, and Jun Xiao. 2019. DEBUG: A Dense Bottom-Up Grounding Approach for Natural Language Video Localization. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*. 5147–5156.
- [45] Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. 2019. ViLBERT: Pretraining Task-Agnostic Visiolinguistic Representations for Vision-and-Language Tasks. In *Advances in Neural Information Processing Systems*. 13–23.
- [46] Huaishao Luo, Lei Ji, Botian Shi, Haoyang Huang, Nan Duan, Tianrui Li, Xilin Chen, and Ming Zhou. 2020. Univilm: A unified video and language pre-training model for multimodal understanding and generation. *arXiv preprint arXiv:2002.06353* (2020).
- [47] Antoine Miech, Jean-Baptiste Alayrac, Lucas Smaira, Ivan Laptev, Josef Sivic, and Andrew Zisserman. 2020. End-to-end learning of visual representations from uncurated instructional videos. In *Proceedings of the IEEE/CVF Conference*

- on *Computer Vision and Pattern Recognition*. 9879–9889.
- [48] Antoine Miech, Ivan Laptev, and Josef Sivic. 2018. Learning a text-video embedding from incomplete and heterogeneous data. *arXiv preprint arXiv:1804.02516* (2018).
- [49] Ishan Misra and Laurens van der Maaten. 2020. Self-supervised learning of pretext-invariant representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 6707–6717.
- [50] Niluthpol Chowdhury Mithun, Juncheng Li, Florian Metzke, and Amit K Roy-Chowdhury. 2018. Learning joint embedding with multimodal cues for cross-modal video-text retrieval. In *Proceedings of the 2018 ACM on International Conference on Multimedia Retrieval*. 19–27.
- [51] Niluthpol Chowdhury Mithun, Sujoy Paul, and Amit K. Roy-Chowdhury. 2019. Weakly Supervised Video Moment Retrieval From Text Queries. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 11592–11601.
- [52] Jonghwan Mun, Minsu Cho, and Bohyung Han. 2020. Local-Global Video-Text Interactions for Temporal Grounding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 10810–10819.
- [53] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. 2018. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748* (2018).
- [54] Yingwei Pan, Tao Mei, Ting Yao, Houqiang Li, and Yong Rui. 2016. Jointly modeling embedding and translation to bridge video and language. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 4594–4602.
- [55] Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global Vectors for Word Representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*. 1532–1543.
- [56] Cristian Rodriguez, Edison Marrese-Taylor, Fatemeh Sadat Saleh, Hongdong Li, and Stephen Gould. 2020. Proposal-free Temporal Moment Localization of a Natural-Language Query in Video using Guided Attention. In *The IEEE Winter Conference on Applications of Computer Vision*.
- [57] Minjoon Seo, Aniruddha Kembhavi, Ali Farhadi, and Hannaneh Hajishirzi. 2017. Bidirectional Attention Flow for Machine Comprehension. In *International Conference on Learning Representations*.
- [58] Dian Shao, Yu Xiong, Yue Zhao, Qingqiu Huang, Yu Qiao, and Dahua Lin. 2018. Find and focus: Retrieve and localize video events with natural language queries. In *Proceedings of the European Conference on Computer Vision (ECCV)*. 200–216.
- [59] Chen Sun, Fabien Baradel, Kevin Murphy, and Cordelia Schmid. 2019. Learning video representations using contrastive bidirectional transformer. *arXiv preprint arXiv:1906.05743* (2019).
- [60] Fan-Yun Sun, Jordan Hoffman, Vikas Verma, and Jian Tang. 2020. InfoGraph: Unsupervised and Semi-supervised Graph-Level Representation Learning via Mutual Information Maximization. In *International Conference on Learning Representations*.
- [61] Hao Tan and Mohit Bansal. 2019. LXMERT: Learning Cross-Modality Encoder Representations from Transformers. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*. 5103–5114.
- [62] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is All you Need. In *Advances in Neural Information Processing Systems*. 5998–6008.
- [63] Petar Veličković, William Fedus, William L. Hamilton, Pietro Liò, Yoshua Bengio, and R Devon Hjelm. 2019. Deep Graph Infomax. In *International Conference on Learning Representations*.
- [64] Subhashini Venugopalan, Marcus Rohrbach, Jeffrey Donahue, Raymond Mooney, Trevor Darrell, and Kate Saenko. 2015. Sequence to sequence-video to text. In *Proceedings of the IEEE international conference on computer vision*. 4534–4542.
- [65] Hao Wang, Zheng-Jun Zha, Xuejin Chen, Zhiwei Xiong, and Jiebo Luo. 2020. Dual Path Interaction Network for Video Moment Localization. In *Proceedings of the 28th ACM International Conference on Multimedia* (Seattle, WA, USA). 4116–4124.
- [66] Jingwen Wang, Lin Ma, and Wenhao Jiang. 2020. Temporally Grounding Language Queries in Videos by Contextual Boundary-aware Prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence*.
- [67] Weining Wang, Yan Huang, and Liang Wang. 2019. Language-Driven Temporal Activity Localization: A Semantic Matching Reinforcement Learning Model. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 334–343.
- [68] Wenhui Wang, Nan Yang, Furu Wei, Baobao Chang, and Ming Zhou. 2017. Gated Self-Matching Networks for Reading Comprehension and Question Answering. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 189–198.
- [69] Jie Wu, Guanbin Li, Xiaoguang Han, and Liang Lin. 2020. Reinforcement Learning for Weakly Supervised Temporal Grounding of Natural Language in Untrimmed Videos. In *Proceedings of the 28th ACM International Conference on Multimedia* (Seattle, WA, USA). 1283–1291.
- [70] Jie Wu, Guanbin Li, Si Liu, and Liang Lin. 2020. Tree-Structured Policy based Progressive Reinforcement Learning for Temporally Language Grounding in Video. In *Proceedings of the AAAI Conference on Artificial Intelligence*.
- [71] Zhirong Wu, Yuanjun Xiong, Stella X Yu, and Dahua Lin. 2018. Unsupervised feature learning via non-parametric instance discrimination. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 3733–3742.
- [72] Ran Xu, Caiming Xiong, Wei Chen, and Jason Corso. 2015. Jointly modeling deep video and compositional text to bridge vision and language in a unified framework. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 29.
- [73] Adams Wei Yu, David Dohan, Quoc Le, Thang Luong, Rui Zhao, and Kai Chen. 2018. Fast and Accurate Reading Comprehension by Combining Self-Attention and Convolution. In *International Conference on Learning Representations*.
- [74] Licheng Yu, Zhe Lin, Xiaohui Shen, Jimei Yang, Xin Lu, Mohit Bansal, and Tamara L Berg. 2018. MAttNet: Modular attention network for referring expression comprehension. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 1307–1315.
- [75] Yitian Yuan, Lin Ma, Jingwen Wang, Wei Liu, and Wenwu Zhu. 2019. Semantic Conditioned Dynamic Modulation for Temporal Sentence Grounding in Videos. In *Advances in Neural Information Processing Systems*. 536–546.
- [76] Yitian Yuan, Tao Mei, and Wenwu Zhu. 2019. To Find Where You Talk: Temporal Sentence Localization in Video with Attention Based Location Regression. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33. 9159–9166.
- [77] Runhao Zeng, Haoming Xu, Wenbing Huang, Peihao Chen, Mingkui Tan, and Chuang Gan. 2020. Dense regression network for video grounding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 10287–10296.
- [78] Bowen Zhang, Hexiang Hu, Joonseok Lee, Ming Zhao, Sheide Chammas, Vihan Jain, Eugene Ie, and Fei Sha. 2020. A Hierarchical Multi-Modal Encoder for Moment Localization in Video Corpus. *arXiv preprint arXiv:2011.09046* (2020).
- [79] Bowen Zhang, Hexiang Hu, and Fei Sha. 2018. Cross-modal and hierarchical modeling of video and text. In *Proceedings of the European Conference on Computer Vision (ECCV)*. 374–390.
- [80] Hao Zhang, Aixin Sun, Wei Jing, Liangli Zhen, Joey Tianyi Zhou, and Rick Siow Mong Goh. 2021. Natural Language Video Localization: A Revisit in Span-based Question Answering Framework. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2021).
- [81] Hao Zhang, Aixin Sun, Wei Jing, and Joey Tianyi Zhou. 2020. Span-based Localizing Network for Natural Language Video Localization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, Online, 6543–6554.
- [82] Songyang Zhang, Houwen Peng, Jianlong Fu, and Jiebo Luo. 2020. Learning 2D Temporal Adjacent Networks for Moment Localization with Natural Language. In *Proceedings of the AAAI Conference on Artificial Intelligence*.
- [83] Songyang Zhang, Jinsong Su, and Jiebo Luo. 2019. Exploiting Temporal Relationships in Video Moment Localization with Natural Language. In *Proceedings of the 27th ACM International Conference on Multimedia* (Nice, France). 1230–1238.
- [84] Zhu Zhang, Zhijie Lin, Zhou Zhao, and Zhenxin Xiao. 2019. Cross-Modal Interaction Networks for Query-Based Moment Retrieval in Videos. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*. 655–664.
- [85] Linchao Zhu and Yi Yang. 2020. ActBERT: Learning Global-Local Video-Text Representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 8746–8755.
- [86] Chengxu Zhuang, Alex Lin Zhai, and Daniel Yamins. 2019. Local aggregation for unsupervised learning of visual embeddings. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 6002–6012.