Text-based Localization of Moments in a Video Corpus

Sudipta Paul, *Student Member, IEEE*, Niluthpol Chowdhury Mithun, *Member, IEEE*, and Amit K. Roy-Chowdhury, *Fellow, IEEE*

***Abstract*—Prior works on text-based video moment localization focus on temporally grounding the textual query in an untrimmed video. These works assume that the relevant video is already known and attempt to localize the moment on that relevant video only. Different from such works, we relax this assumption and address the task of localizing moments in a corpus of videos for a given sentence query. This task poses a unique challenge as the system is required to perform: (i) retrieval of the relevant video where only a segment of the video corresponds with the queried sentence, and (ii) temporal localization of moment in the relevant video based on sentence query. Towards overcoming this challenge, we propose Hierarchical Moment Alignment Network (HMAN) which learns an effective joint embedding space for moments and sentences. In addition to learning subtle differences between intra-video moments, HMAN focuses on distinguishing inter-video global semantic concepts based on sentence queries. Qualitative and quantitative results on three benchmark text-based video moment retrieval datasets - Charades-STA, DiDeMo, and ActivityNet Captions - demonstrate that our method achieves promising performance on the proposed task of temporal localization of moments in a corpus of videos.**

***Index Terms*—Temporal Localization, Video Moment Retrieval, Video Corpus**

I. Introduction

Localizing activity moments in long and untrimmed videos is a prominent video analysis problem. Early works on moment localization were mostly limited by the use of a predefined set of labels to describe an activity [1], [2], [3], [4]. However, due to the nature of the complexity of real-life activities, natural language sentences would be the appropriate choice to describe an activity rather than a predefined set of labels. Recently, there are several works [5], [6], [7], [8], [9], [10], [11], [12], [13], [14] that utilize sentence queries to temporally localize moments in untrimmed videos. All these approaches build upon an underlying assumption that the correspondence between sentences and videos is known. As a result, these approaches attempt to localize moments only in the related video. We argue that such an assumption of knowing relevant videos a priori may not be plausible for most practical scenarios. It is more likely that a user would need to retrieve a moment from a large corpus of videos given a sentence query.

In this work, we relax the assumption of specified video-sentence correspondence of the prior works on temporal moment localization and address the more challenging task of localizing moments in a corpus of videos. For example in Figure 1, the moment marked by the green dashed box in video-*(b)* corresponds to the text query: *'Person puts clothes into a washing machine'*. Prior works on temporal moment localization only attempt to detect the temporal endpoints in the given video-*(b)* by learning to identify subtle changes in dynamics of the activity. However, the task of localizing the correct moment in the illustrated collection of videos (i.e., *(a)*, *(b)*, and *(c)* in Figure 1) imposes the additional requirement to distinguish moments from different videos and identify the correct video (video-*(b)*) based on the differences of putting and pulling activities as well as the presence of washing machine and clothes.

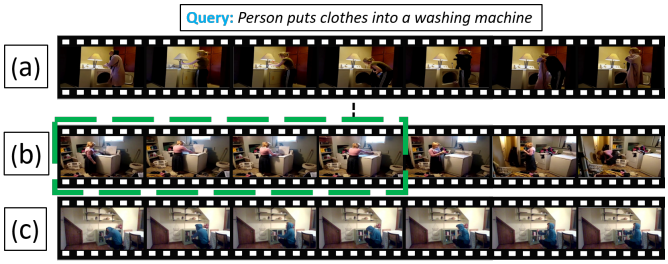

Fig. 1. Example illustration of our proposed task. We consider localizing moments in a corpus of videos given a text query. Here, for the queried text: 'Person puts clothes into a washing machine', the system is required to identify the relevant video-*(b)* from the illustrated corpus of videos (video-*(a)*, video-*(b)*, and video-*(c)*) and temporally localize the pertinent moment (ground truth moment marked by the green dashed box) in that relevant video.

To address this problem, a trivial approach would be to use an off-the-shelf video-text retrieval module to retrieve the relevant video and then localize the moment in that retrieved video. Most of the video-text retrieval approaches [15], [16], [17], [18], [19], [20], [21], [22] are designed for cases where videos and text queries have a one-to-one correspondence, i.e., a query sentence reflects a trimmed and short video or a query paragraph represents a long and untrimmed video. However, in our addressed task, the query sentence reflects a segment of a long and untrimmed video, and different segments of a video can be associated with different language annotations, resulting in one-to-many video-text correspondence. Hence, the existing video-text retrieval approaches are likely to fall short on our target task. Another trivial approach would be to scale up the temporal localization of moments approaches, i.e., instead of searching over a given video, it searches over the corpus of videos. However, these approaches are only designed to discern intra-video moments based on sentence semantics

• Sudipta Paul, and Amit K. Roy-Chowdhury are with the Department of Electrical and Computer Engineering, University of California, Riverside, CA, USA. Niluthpol Chowdhury Mithun is with SRI International, Princeton, NJ, USA. E-mails: (spaul007@ucr.edu, niluthpol.mithun@sri.com, amitrc@ece.ucr.edu)

and fail to distinguish moments from different videos and identify the correct video.

In this work, based on the text query, we focus on discerning moments from different videos as well as understand the nuances of activities simultaneously to localize the correct moment in the relevant video. Our objective is to learn a joint embedding space that will align representations of corresponding video moments and sentences. For this, we propose **H**ierarchical **M**oment **A**lignment **N**etwork (**HMAN**), a novel neural network framework that effectively learns a joint embedding space to align corresponding video moments and sentences. Learning joint embedding space for retrieval or localization tasks has been addressed by several other methods [6], [23], [22], [24], [25], [26]. Among them, [6] and [23] are closely related to our work as they try to align corresponding moment and sentence representations in the joint embedding space. However, our approach is significantly different from these works. In contrast to these works, HMAN utilizes temporal convolutional layers in a hierarchical structure to represent candidate video moments. It allows the model to generate all candidate moment representations of a video in a single pass, which is more efficient than sliding based approaches like [6], [23]. Our learning objective is also different from [6], [23], where they only try to distinguish between intra-video moments and inter-video moments. In our proposed approach, in addition to distinguishing intra-video moments, we propose a novel learning objective that utilizes text-guided global semantics to distinguish different videos. Global semantics of a video refers to the semantics that is common across most of the moments of that video. As the global semantics vary across videos, by distinguishing videos, we learn to distinguish inter-video moments. We demonstrate the advantage of our proposed approach over other baseline approaches and contemporary works on three benchmark datasets.

A. Contributions

The main contributions of the proposed work are as follows:

- We explore an important, yet under-explored, problem of text query-based localization of moments in a video corpus.
- We propose a novel framework, HMAN, that uses stacked temporal convolutional layers in a hierarchical structure to represent video moments and texts jointly in an embedding space. Combined with the proposed learning objective, the model is able to align moment and sentence representations by distinguishing both local subtle differences of the moments as well as global semantics of the videos simultaneously.
- Towards solving the problem, we propose a novel learning objective that utilizes text-guided global semantics of the videos to distinguish moments from different videos.
- We empirically show the efficacy of our proposed approach on DiDeMo, Charades-STA, and ActivityNet Captions dataset and study the significance of our proposed learning objective.

II. Related Works

Video-Text Retrieval. Among the cross-modal retrieval tasks [27], [28], [29], [30], [31], video-text retrieval has gained much attention recently. Emergence of datasets like the Microsoft Research Video to Text (MSR-VTT) [32], the MPII movie description dataset as part of the Large Scale Movie Description Challenge (LSMDC) dataset [33], and Microsoft Video Description Dataset (MSVD) [34] have boosted video-text retrieval task. These datasets contain short video clips with accompanying natural language. Initial approaches for the video-text retrieval task were based on concept classification [35], [36], [37]. Recent approaches focus on directly encoding video and text in a common space and retrieving relevant instances based on some similarity measure in the common space [30], [31], [38], [39], [40], [41]. These works used Convolutional Neural Network (CNN) [39] or Long Short-Term Memory Network (LSTM) [42] for video encoding. To encode text representations, Recurrent Neural Network (RNN) [38], bidirectional LSTM [39] and GRU [16] were commonly used. Mithun et al. [16] employed multimodal cues such as image, motion, and audio for video encoding. In [19], multi-level encodings for video and text were used and both videos and sentences were encoded in a similar manner. Liu et al. [43] proposed collaborative experts model to aggregate information effectively from different pre-trained experts. Yu et al. [39] proposed a Joint Sequence Fusion model for sequential interaction of videos and texts. Song et al. [44] introduced Polysemous Instance Embedding Networks that compute multiple and diverse representations of an instance. Among the recent works, Wray et al. [18] enriched the embedding learning by disentangling parts-of-speech of captions. Chen et al. [45] used Hierarchical Graph Reasoning to improve fine-grained video-text retrieval. Another line of work considers video-paragraph retrieval. For example, Zhang et al. [15] proposed hierarchical modeling of videos, and paragraphs and Shao et al. [17] utilized top-level and part-level association for the task of video-paragraph retrieval. However, all of these approaches have an underlying assumption that videos and text queries have one-to-one correspondence. As a result, they are not adaptable for our addressed task, where the video-text pairs have one-to-many correspondence.

Temporal Localization of Moments. The task of localizing a moment/activity in a given long and untrimmed video via text query was introduced in [5], [6]. After that, there have been a lot of works [7], [8], [9], [10], [11], [12], [13], [46], [47], [48], [49], [50], [51], [52], [53], [54], [55], [56], [57], [58] that addressed this task. All of these works on temporal localization of moments can be divided into two categories: i) two stage approaches that sample segments of videos in the first step and then try to find a semantic alignment between sentences and those segments of videos in the second step [5], [6], [7], [8], [9], [10], [11], and ii) single stage approaches that predict the association of sentences with multi-scale visual representation units as well as predict temporal boundary for each visual representation unit in a single pass [12], [13]. Among all the approaches, Gao et al. [5] developed Cross-modal Temporal Regression Localizer that jointly models text queries and video clips. A common embedding space for video temporal context features and language features was learnt in [6]. Some of the works focused on vision-language fusion techniques to improve localization performance. For

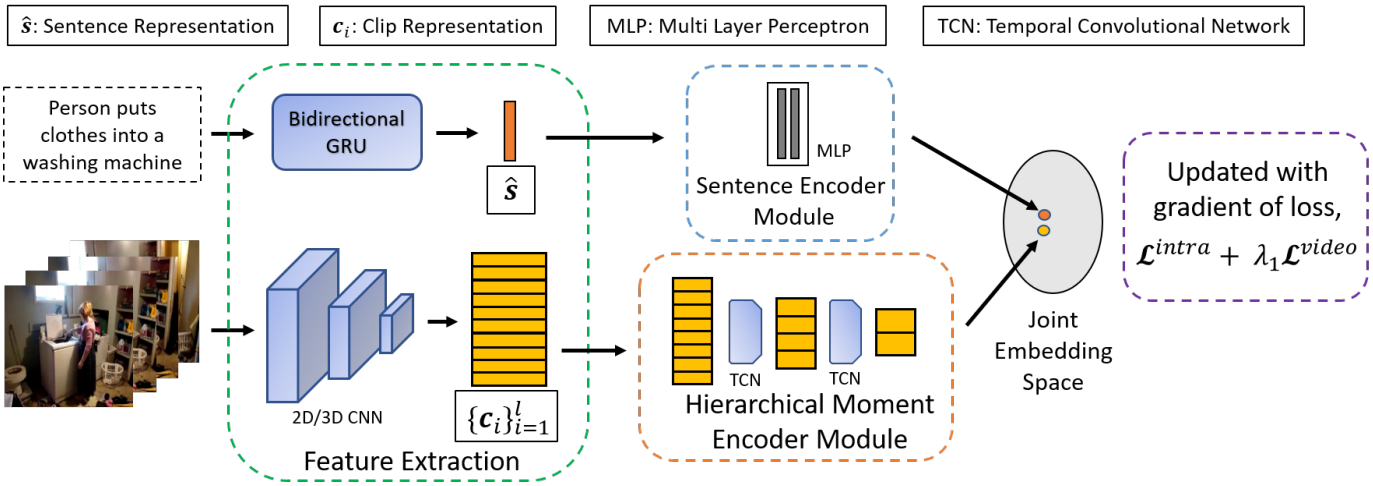

Fig. 2. A brief illustration of the proposed Hierarchical Moment Alignment Network for the moment localization task in a video corpus. The framework uses the feature extraction unit to extract clip and sentence features. Hierarchical moment encoder module and sentence encoder module projects moment representations and sentence representations in the joint embedding space respectively. The network learns to align moment-sentence pairs in the joint embedding space by explicitly focusing on distinguishing intra-video moments and inter-video global semantic differences. (Details of the learning procedure in section III-F)

example, Multimodal Circulant Fusion was incorporated in [7]. Liu et al. [8] incorporated a memory attention mechanism to emphasize the visual features mentioned in the query and simultaneously use their context. Ge et al. [10] mined activity concepts from both video and language modalities to improve the regression performance. Chen et al. [9] proposed Temporal GroundNet which captures evolving fine-grained frame-by-word interactions. Xu et al. [11] used early integration of vision and language for proposal generation and query sentence modulation using visual features. Among the single shot approaches, candidate moment encoding and temporal structural reasoning were unified in a single shot framework in [12]. Semantic Conditioned Dynamic Modulation (SCDM) was proposed in [13] for correlating sentence and related video contents. These approaches on moment localization in a given video show promise, but fall short on realizing the requirement of identifying the correct video to address the task of moment localization in a corpus of videos.

There has been one concurrent work [23] that addressed the task of temporal localization of moments in a video corpus. They adopted the approach of Moment Context Network [6]. However, instead of directly learning moment-sentence alignment as in [6], they tried to learn clip-sentence alignment for scalability issues where a moment consists of multiple clips. Even so, a referring event is likely to consist of multiple clips, and a single clip can not reflect the complete dynamics of an event. Hence, consecutive clips with different contents need to be aligned with the same sentence which results in suboptimal representation for both the clips and the sentence. We later empirically show that our approach is significantly more effective than [23] in the addressed task.

III. Methodology

In this section, we present our framework for the task of text-based temporal localization of moments in a corpus of untrimmed and unsegmented videos. First, we define the problem and provide an overview of the HMAN framework. Then, we present how clip-level video representations and word-level sentence representations are extracted. Then, we describe the framework in detail along with the hierarchical temporal convolutional network to generate moment embeddings and sentence embeddings. Finally, we describe how we learn to encode moment and sentence representations in the joint embedding space for effective retrieval of the moment based on a text query.

A. Problem Statement

Consider that we have a set of N long and untrimmed videos $\mathcal{V} = \{v_i\}_{i=1}^{N}$, where a video v is associated with m_v temporal sentence annotations $\mathcal{T} = \{(s_j, \tau_j^s, \tau_j^e)\}_{j=1}^{m_v}$. Here, s_j is the sentence annotation and τ_j^s, τ_j^e are the starting time and ending time of the moment in the video that corresponds with the sentence annotation s_j. The set of all temporal sentence annotations is $\mathcal{S} = \{\mathcal{T}_i\}_{i=1}^{N}$. Given a natural language query s, our task is to predict a set $s_{det} = \{v, \tau^s, \tau^e\}$ where, v is the video that contains the relevant moment and τ^s, τ^e are the temporal information of that moment.

B. Framework Overview

Our goal is to learn representations for candidate moments and sentences in such a way that the related moment-sentence pairs are aligned in the joint embedding space. Towards this goal, we propose HMAN, which is illustrated in Figure 2. First, we employ a feature extraction unit to extract clip level features $\{\boldsymbol{c}_i\}_{i=1}^{l}$ from a video and sentence features $\hat{\boldsymbol{s}}$ from a sentence. Clip representations and sentence representations are used to learn the semantic alignment between sentences and candidate moments. To project the moment representations and sentence representations in the joint embedding space, we use a hierarchical moment encoder module and a sentence encoder module respectively. The moment encoder module is inspired by single shot temporal action detection approach [4] where temporal convolutional layers are stacked in a hierarchical

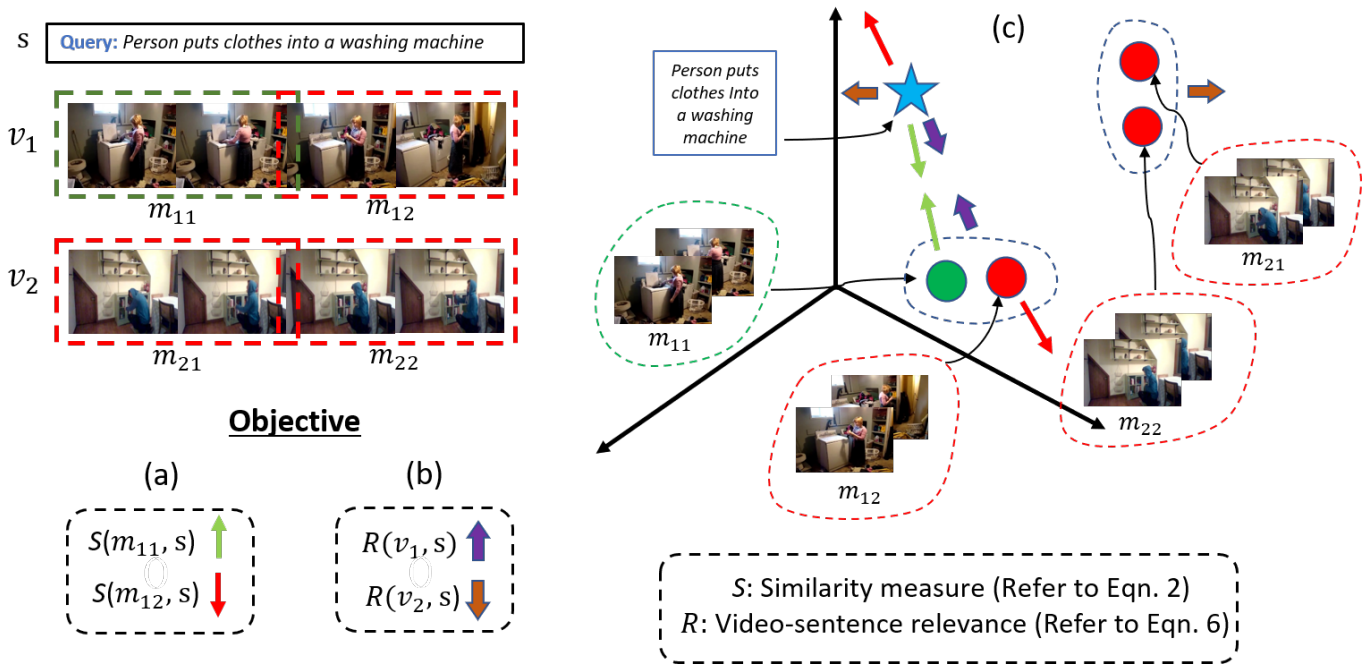

Fig. 3. A conceptual representation of our proposed learning objective. For a text query s with relevant moment m_{11} in a set of videos $\{v_1, v_2\}$ with set of moments $\{m_{11}, m_{12}, m_{21}, m_{22}\}$, we learn the joint embedding space using- (a) intra-video moments: increasing similarity for relevant pair (m_{11}, s) and decreasing similarity for non-relevant pair (m_{12}, s) from the same video, (b) global semantics of video: increasing video-sentence relevance for relevant pair (v_1, s) and decreasing for non-relevant pair (v_2, s), where the video-sentence relevance is computed in terms of moment-sentence similarity. This is also illustrated in (c), where the arrows indicate which pairs are learning to increase their similarity (moving close in the embedding space) and which pairs are learning to decrease their similarity (moving further away in the embedding space). Details can be found in section III-F

structure to obtain multi-scale moment features representing video segments of different duration. For the sentence encoder module, we use a two-layer feedforward neural network. Based on text queries, we derive the learning objective to explicitly focus on distinguishing intra-video moments and inter-video global semantics. We adopted sum-margin based triplet loss [59] and max-margin based triplet loss [59] separately in two different settings to train the model in an end-to-end fashion and gained performance improvement over baseline approaches in both setups. In the inference stage, for a query sentence, the candidate moment with the most similar representation is retrieved from the corpus of videos.

C. Feature Extraction Unit

To work with data from different modalities, we extract feature representations using modality specific pretrained models.

Video Feature Extraction. We extract high level video features using a deep convolutional neural network. Each video v is divided into a set of l non-overlapping clips and we extract features for each clip. As a result, the video is represented by a set of features $\{\boldsymbol{c}_i\}_{i=1}^{l}$, where $\boldsymbol{c}_i$ is the feature representation of the i^{th} clip. To generate representations for all the candidate moments of a video in a single shot approach [4], we keep the input video length, i.e., number of clips, l, fixed. A video longer than the fixed length is truncated and a video shorter than the fixed length is padded with zeros.

Sentence Feature Extraction. To represent sentences, we use GloVe word embedding [60] for each word in a sentence. Then these word embedding sequences are encoded using a Bi-directional Gated Recurrent Unit (GRU) [61] with 512 hidden states. Here, words in a sentence are represented by a 512-dimensional vector, corresponding to their GRU hidden states. So, we can have a set of word-by-word representations of a sentence $S = \{\boldsymbol{h}_i\}_{i=1}^{n}$, where n is the number of words present in the sentence. The average of the word representations is used as the sentence representation $\hat{\boldsymbol{s}}$.

D. Moment Encoder Module

Existing approaches for moment localization based on learning joint visual-semantic embedding space either use a temporal sliding window with multiple scales [6] or optimize over a predefined set of consecutive clips based on clip-sentence similarity [23] to generate candidate segments. However, sliding over a video with different scales or optimizing for all possible combinations of clips is computationally heavy. Again, in both cases, extracted candidate moments or predefined clips are projected in the joint embedding space independent of neighboring or overlapping moments/clips of the same video. Consequently, while learning the moment-sentence or clip-sentence semantic alignment, representations for neighboring or overlapping moments are not constrained to be well clustered to preserve the semantic similarity. Therefore, instead of projecting representations for candidate moments independently and inefficiently in the joint embedding space, inspired by the single shot activity detection [4], we use temporal convolutional layers [62] in a hierarchical setup to project representations for all candidate moments of a video simultaneously. We use a stack of $1D$ convolutional

Algorithm 1 Learning optimized HMAN (max-margin case)

Input: Untrimmed video set $\mathcal{V}$, Temporal sentence annotation set $\mathcal{S}$, Initialized HMAN weights θ
for t = 1 to maxIter **do**
step 1: Construct minibatch of video-sentence pairs
step 2: Extract video and sentence feature
step 3: Project candidate moment and sentence representations in the joint embedding space
step 4: Construct triplets
step 5: Compute $\mathcal{L}_{max}^{intra}$ and $\mathcal{L}_{max}^{video}$ using Eqn. 5 & 10
step 6: Optimize θ by minimizing total loss
end for
Output: Optimized HMAN weights θ

layers where the convolution operation can be denoted as $Conv(\sigma_k, \sigma_s, d)$. Here, σ_k, σ_s, and d indicate the kernel size, stride size, and filter numbers, respectively. The set of moment representations generated for K layers of hierarchical structure is $\{\{\boldsymbol{m}_i^k\}_{i=1}^{T_k}\}_{k=1}^{K}$. Here, T_k is the temporal dimension of the k^{th} layer, which decreases in the following layers. $\boldsymbol{m}_i^k \in \mathcal{R}^d$ is the i^{th} moment representation of the k^{th} layer and k^{th} layer generates T_k moment representations. Feature representations in the top layers of the hierarchy correspond to moments with shorter temporal duration, while the feature representations in the bottom layers correspond to moments with longer duration in a video. We keep the feature dimension of each moment representation fixed to d for all the layers of the temporal convolutional network.

E. Sentence Encoder Module

We learn to project the textual representations in the joint embedding space keeping the inputs from different modalities with similar semantics close to each other. We use two layers of feedforward neural network with learnable parameters $\boldsymbol{W}_1^s$, $\boldsymbol{W}_2^s$, $\boldsymbol{b}_1^s$, and $\boldsymbol{b}_2^s$ to project the sentence representation $\hat{\boldsymbol{s}}$ in the joint embedding space, which can be defined as,

$$\boldsymbol{s} = \boldsymbol{W}_2^s\Big(BN\big(ReLU(\boldsymbol{W}_1^s\hat{\boldsymbol{s}} + \boldsymbol{b}_1^s)\big)\Big) + \boldsymbol{b}_2^s \tag{1}$$

Here, the dimension of the projected sentence representation $\boldsymbol{s}$ is kept consistent with the projected moment representation $\boldsymbol{m}$ ($\boldsymbol{m}, \boldsymbol{s} \in \mathcal{R}^d$).

F. Learning Joint Embedding Space

Projected representations in the joint embedding space from different modalities need to be close to each other if they are semantically related. Training procedures to learn projecting representations in the joint embedding space mostly adopts two common loss functions: sum-margin based triplet ranking loss [59] and max-margin based triplet ranking loss [63]. We consider both of these loss functions separately. As illustrated in Figure 3, we focus on distinguishing intra-video moments and inter-video global semantic concepts. In this section, we discuss our approach to learn projecting representations from different modalities in the joint embedding space for multimodal data.

Similarity Measure. We use the cosine similarity of projected representations from two modalities in the joint embedding space to infer their semantic relatedness. So, the similarity between a candidate moment m and a sentence s is,

$$S(m, s) = \frac{\boldsymbol{m}^T\boldsymbol{s}}{\|\boldsymbol{m}\|\|\boldsymbol{s}\|} \tag{2}$$

where $\boldsymbol{m}$ and $\boldsymbol{s}$ are the projected moment representation and sentence representation in the joint embedding space.

Learning for Intra-video Moments. To localize a sentence query in a video, the model needs to identify the subtle differences of the candidate moments from the same video and distinguish them. Among the candidate segments of a video, one or few of the moments can be considered related to the query sentence based on some IoU threshold. While training the network, we consider related moments with the queried sentence as the positive pairs and non-corresponding moments with the queried sentence as the negative pairs. Suppose, for a pair of video-sentence (v, s), we consider the set of positive moment-sentence pairs $\{(m, s)\}$ and the set of negative moment-sentence pairs $\{(m^-, s)\}$. We compute the intra-video ranking loss for all video-sentence pairs $\{(v, s)\}$. Using the sum-margin setup, the intra-video triplet loss is:

$$\mathcal{L}_{sum}^{intra} = \sum_{\{(v,s)\}}\sum_{\{(m,s)\}}\sum_{\{(m^-,s)\}}\big[\alpha_{intra} - S(m,s) + S(m^-,s)\big]_+ \tag{3}$$

Similarly, using the max-margin setup, we calculate the intra-video triplet loss by,

$$\hat{m} = \arg\max_{m^-} S(m^-, s) \tag{4}$$

$$\mathcal{L}_{max}^{intra} = \sum_{\{(v,s)\}}\sum_{\{(m,s)\}}\big[\alpha_{intra} - S(m,s) + S(\hat{m},s)\big]_+ \tag{5}$$

Here, $[f]_+ = max(0, f)$ and α_{intra} is the ranking loss margin for intra-video moments.

Learning for Videos. Learning to distinguish intra-video moments only allows the model to learn subtle changes in the video. It does not allow the model to distinguish moments from different videos. However, learning to differentiate moments from different videos is important as we need to localize the correct moment in the video corpus. Hence, we also learn to distinguish moments from different videos by capitalizing on the text-guided global semantics of videos. As the global semantics varies across videos we try to distinguish videos based on these global semantics. To do so, we learn to maximize the relevance of correct video-sentence pairs. Video-sentence relevance is computed in terms of moment-sentence relevance. As a result, learning to align video-sentence pairs enforces constraints on the representation of moments from different videos to be dissimilar. Inspired by the work of [27], we compute the relevance of a video and a sentence by,

$$R(v, s) = \log\Big(\sum_{\{m\}}\exp\big(\beta S(m, s)\big)\Big)^{1/\beta}, \tag{6}$$

where β is a factor that determines how much to magnify the importance of the most relevant moment-sentence pair and

TABLE I
TABULATED SUMMARY OF THE DETAILS OF DATASET CONTENTS

Dataset	Number of videos		Moment-sentence pairs
	Total	Train/Val/Test	
DiDeMo	10464	8395 / 1065 / 1004	26892
Charades-STA	6670	5336 / - / 1334	16128
ActivityNet Captions	20k	10009 / 4917 / -	71942

TABLE II
TABULATED SUMMARY OF THE IMPLEMENTATION DETAILS REGARDING VIDEO PROCESSING FOR THREE DATASETS

Dataset	Video length	# of candidate moments	Per Unit duration	Temporal dimension of layers
DiDeMo	12	21	2.5s	{6,5,4,3,2,1}
Charades-STA	64	61	1s	{31,16,8,4,2,1}
ActivityNet Captions	512	1023	1s	{512, 256, 128, 64, 32, 16, 8, 4 ,2, 1}

$\{m\}$ is the set of all the moments in video v. As $\beta \to \infty$, $R(v,s)$ approximates $\max_{m_i \in v} S(m_i, s)$. This is necessary because all the segments of the video do not correspond to the sentence.

For each positive video-sentence pair (v,s) where the sentence s relates to a segment of the video v, we can consider two sets of negative pairs $\{(v^-,s)\}$ and $\{(v,s^-)\}$. Using the sum-margin setup, we calculate the triplet loss for video-sentence alignment of all the positive video-sentence pairs $\{(v,s)\}$ by,

$$\mathcal{L}_{sum}^{video} = \sum_{\{(v,s)\}} \sum_{\{(v^-,s)\}} \big[\alpha_{video} - R(v,s) + R(v^-,s)\big]_+ + \sum_{\{(v,s)\}} \sum_{\{(v,s^-)\}} \big[\alpha_{video} - R(v,s) + R(v,s^-)\big]_+ \quad (7)$$

Similarly, using the max-margin setup, we compute the triplet loss for video-sentence alignment by,

$$\hat{v} = \arg\max_{v^-} R(v^-,s) \quad (8)$$

$$\hat{s} = \arg\max_{s^-} R(v,s^-) \quad (9)$$

$$\mathcal{L}_{max}^{video} = \sum_{\{(v,s)\}} \big[\alpha_{video} - R(v,s) + R(\hat{v},s)\big]_+ + \sum_{\{(v,s)\}} \big[\alpha_{video} - R(v,s) + R(v,\hat{s})\big]_+ \quad (10)$$

Here, α_{video} is the ranking loss margin for learning inter-video global semantic concepts.

Learning Objective. We combine the calculated loss for intra-video case and video-sentence alignment case and try to minimize it as our final objective. For the sum-margin setup, the final objective is,

$$\min_{\theta} \mathcal{L}_{sum}^{intra} + \lambda_1 \mathcal{L}_{sum}^{video} + \alpha \|\mathcal{W}\|_F^2 \quad (11)$$

Similarly, for the max-margin setup, the final objective is,

$$\min_{\theta} \mathcal{L}_{max}^{intra} + \lambda_1 \mathcal{L}_{max}^{video} + \alpha \|\mathcal{W}\|_F^2 \quad (12)$$

Here, θ represents the network weights and all the learnable weights are lumped together in $\mathcal{W}$. λ_1 balances the contribution between learning to distinguish intra-video moments and learning to distinguish videos based on a text query. α is the weight on the regularization loss. Our objective is to optimize θ to generate a proper representation for candidate moments and sentences to minimize these combined losses. During training, these losses are computed for a mini-batch where the mini-batches are sampled randomly from the entire training set. This stochastic approach yields the advantage of reducing the probability of selecting instances with high semantic relation as the negative samples.

Inference. In the inference step, for a query sentence, we compute the similarity of candidate moment representations with the query sentence representation using Eqn. 2. We retrieve the candidate moment from the video corpus that results in the highest similarity.

IV. EXPERIMENTS

In this section, we experimentally evaluate the performance of our proposed method for the task of temporal localization of moments in a corpus of video. We first discuss the datasets we use and the implementation details of the experiments. Then we report and analyze the results both quantitatively and qualitatively.

A. Datasets

We conduct experiments and evaluate the performance on three benchmark text-based video moment retrieval datasets, namely DiDeMo [6], Charades-STA [5], and ActivityNet Captions [66]. All of these datasets contain unsegmented and untrimmed videos with natural language sentence annotations with temporal information. Table I summarizes the details of the contents of three datasets.

DiDeMo. The Distinct Describable Moments (DiDeMo) dataset [6] is one of the most diverse datasets for the temporal localization of moments in videos given natural language descriptions. The videos are collected from Flickr and each video is trimmed to a maximum of 30 seconds. The videos in the dataset are divided into 5-second segments to reduce the complexity of annotation. The dataset is split into training, validation, and test sets containing 8,395, 1,065, and 1,004 videos respectively. The dataset contains a total of 26,892 moment-sentence pairs and each natural language description is temporally grounded by multiple annotators.

Charades-STA. Charades-STA dataset is introduced in [5] to address the task of temporal localization of moments in untrimmed videos. The dataset contains a total of 6,670 videos with 16,128 moment-sentence pairs. We have used the published split of videos during training and testing (train-5,336, test-1,334). As a result, the training set and the testing set contain 12,408 and 3,720 moment-sentence pairs respectively. This dataset is originally built on the Charades [67] activity dataset with temporal activity annotation and video-level description. Authors in [5] adopted a keyword matching strategy to generate clip-level sentence annotation.

ActivityNet Captions. ActivityNet Captions [66] dataset, which is proposed for dense video captioning task, is built

TABLE III
COMPARISON OF PERFORMANCE FOR THE TASK OF TEMPORALLY LOCALIZING MOMENTS IN A VIDEO CORPUS ON DIDEMO DATASET. († REPORTED FROM [23]) (↓ INDICATES THE PERFORMANCE IS BETTER IF THE SCORE IS LOW)

	Feature used	DiDeMo					
		$IoU = 0.50$			$IoU = 0.70$		
		R@10	R@100	MR↓	R@10	R@100	MR↓
Moment Prior† [23]	-	0.22	2.34	2527	0.17	1.99	3234
MCN† [6]	RGB (ResNet-152)	2.15	12.47	1057	1.55	9.03	1423
SCDM [13]	RGB (ResNet-152) + Flow (TSN)	0.57	4.43	-	0.22	1.42	-
VSE++ [63] + SCDM [13]	RGB (ResNet-152) + Flow (TSN)	0.70	4.16	-	0.30	2.81	-
CAL† [23]	RGB (ResNet-152)	3.90	16.51	831	2.81	12.79	1148
HMAN (sum-margin, Eqn. 11)	RGB (ResNet-152)	5.63	26.49	412	4.51	20.82	546
HMAN (TripSiam [64])	RGB (ResNet-152) + Flow (TSN)	2.34	17.82	509	1.59	13.92	637
HMAN (DSLT [65])	RGB (ResNet-152) + Flow (TSN)	5.95	25.45	313	4.66	20.04	447
HMAN (sum-margin, Eqn. 11)	RGB (ResNet-152) + Flow (TSN)	**6.25**	**28.39**	**302**	**4.98**	**22.51**	**416**
HMAN (max-margin, Eqn. 12)	RGB (ResNet-152) + Flow (TSN)	5.47	20.82	618	3.86	16.28	905

TABLE IV
COMPARISON OF PERFORMANCE FOR THE TASK OF TEMPORALLY LOCALIZING MOMENTS IN A VIDEO CORPUS ON CHARADES-STA DATASET. († REPORTED FROM [23]) (↓ INDICATES THE PERFORMANCE IS BETTER IF THE SCORE IS LOW)

	Feature used	Charades-STA					
		$IoU = 0.50$			$IoU = 0.70$		
		R@10	R@100	MR↓	R@10	R@100	MR↓
Moment Prior† [23]	-	0.17	1.63	4906	0.05	0.56	11699
MCN† [6]	RGB (ResNet-152)	0.52	2.96	6540	0.31	1.75	10262
SCDM [13]	RGB (I3D)	0.73	6.41	-	0.56	4.23	-
VSE++ [63] + SCDM [13]	RGB (I3D)	1.02	5.06	-	0.70	3.37	-
CAL† [23]	RGB (ResNet-152)	0.75	4.39	5486	0.42	2.78	8627
HMAN (TripSiam [64])	RGB (I3D)	1.27	7.60	2821	0.70	4.49	5766
HMAN (DSLT [65])	RGB (I3D)	1.05	7.27	2390	0.54	4.61	**5496**
HMAN (sum-margin, Eqn. 11)	RGB (I3D)	1.29	7.73	2418	0.83	4.12	6395
HMAN (max-margin, Eqn. 12)	RGB (I3D)	**1.40**	**7.79**	**2183**	**1.05**	**4.69**	5812

on the ActivityNet dataset [68]. It consists of YouTube video footage where each video contains at least two ground truth segments and each segment is paired with one ground truth caption [11]. This dataset contains around 20k videos which are split into training, validation, and testing set. We use the published splits over videos (train set – 10,009 videos, validation set – 4,917 videos), where the evaluation is done on the validation set. Videos are typically longer in length than DiDeMo and Charades-STA datasets.

B. Evaluation Metric

We use the standard evaluation criteria adopted by various previous temporal moment localization works [5], [13], [12]. These works use $R@k, IoU{=}m$ metric, which reports the percentage of cases where at least one of the top-k results have Intersection-over-Union (IoU) larger than m [5]. For a sentence query, $R@k, IoU{=}m$ reflects if one of the top-k retrieved moments has Intersection-over-Union with the ground truth moment larger than the specified threshold m. So, for each query sentence, $R@k, IoU{=}m$ is either 1 or 0. As this metric is associated with a queried sentence, we compute it for all the sentence queries in the testing set (DiDeMo, Charades-STA) or in the validation set (ActivityNet Captions) and report the average results. We report $R@k, IoU{=}m$ over all queried sentences for $k \in \{10, 100\}$ and $m \in \{0.50, 0.70\}$. We also use median retrieval rank (MR) as an evaluation metric. MR computes the median of the rank of the correct moment for each query. Lower values of MR indicate good performance. We compute MR for $IoU \in \{0.50, 0.70\}$. Note that DiDeMo dataset provides multiple temporal annotations for each sentence. We consider a detection is correct if it overlaps with a minimum of two temporal annotations with a specified IoU.

TABLE V
COMPARISON OF PERFORMANCE FOR THE TASK OF TEMPORALLY LOCALIZING MOMENTS IN A VIDEO CORPUS ON ACTIVITYNET CAPTIONS DATASET. († REPORTED FROM [23])

	Feature used	ActivityNet Captions			
		$IoU = 0.50$		$IoU = 0.70$	
		R@10	R@100	R@10	R@100
Moment Prior†	-	0.05	0.47	0.03	0.26
MCN† [6]	RGB (ResNet-152)	0.18	1.26	0.09	0.70
CAL† [23]	RGB (ResNet-152)	0.21	1.58	0.10	0.90
HMAN (sum)	RGB (C3D)	0.43	2.84	0.22	1.48
HMAN (max)	RGB (C3D)	**0.66**	**4.75**	**0.32**	**2.27**

C. Implementation Details

For DiDeMo dataset, we use ResNet-152 features [69], where pool5 features are extracted at 5 fps over the video frames. Then the features are max-pooled over $2.5s$ clips. Also, we extract optical flow features from the penultimate

TABLE VI
COMPARISON OF THE PERFORMANCE OF HMAN WITH/WITHOUT THE HIERARCHICAL MOMENT ENCODER MODULE. THE EXPERIMENTS ARE DONE FOR DIDEMO AND CHARADES-STA DATASETS. († REPORTED FROM [23]) (↓ INDICATES THE PERFORMANCE IS BETTER IF THE SCORE IS LOW)

	DiDeMo						Charades-STA					
	$IoU = 0.50$			$IoU = 0.70$			$IoU = 0.50$			$IoU = 0.70$		
	R@10	R@100	MR↓	R@10	R@100	MR↓	R@10	R@100	MR↓	R@10	R@100	MR↓
HMAN (sum, w/o TCN)	3.44	14.14	1168	2.14	9.91	1636	1.13	6.12	4170	0.43	4.09	8295
HMAN (sum, w/ TCN)	**6.25**	**28.39**	**302**	**4.98**	**22.51**	**416**	**1.29**	**7.73**	**2418**	**0.83**	**4.12**	**6395**
HMAN (max, w/o TCN)	3.41	12.13	1603	1.99	8.96	2214	0.70	4.71	5800	0.46	3.13	10907
HMAN (max, w/ TCN)	**5.47**	**20.82**	**618**	**3.86**	**16.28**	**905**	**1.40**	**7.79**	**2183**	**1.05**	**4.69**	**5812**

TABLE VII
ABLATION STUDY FOR THE EFFECTIVENESS OF LEARNING EMBEDDING SPACE UTILIZING DIFFERENT LOSS COMPONENTS AS DESCRIBED IN III-F FOR DIDEMO DATASET USING SUM-MARGIN SET UP.

	$IoU = 0.50$		$IoU = 0.70$	
	R@10	R@100	R@10	R@100
HMAN (intra)	0.57	6.00	0.52	4.71
HMAN (video)	1.77	10.03	0.30	2.34
HMAN (proposed)	**6.25**	**28.39**	**4.98**	**22.51**

TABLE VIII
PERFORMANCE COMPARISON FOR THE TASK OF RETRIEVING CORRECT VIDEO BASED ON SENTENCE QUERY ON DIDEMO AND CHARADES-STA DATASET.

	DiDeMo			Charades-STA		
	R@10	R@100	R@200	R@10	R@100	R@200
VSE++ [63]	2.49	16.81	29.53	1.89	13.31	24.43
HMAN (max)	12.43	42.43	58.22	2.26	15.87	27.26
HMAN (sum)	**15.36**	**55.23**	**69.12**	**2.45**	**18.51**	**30.52**

layer from a competitive activity recognition model [70]. We use Kinetics pretrained I3D network [71] to extract per second clip features for the Charades-STA dataset. For ActivityNet Captions dataset, we use extracted C3D features [72]. We set the number of input clips of a video, $l = 12$ for DiDeMo dataset, $l = 64$ for Charades-STA dataset, and $l = 512$ for ActivityNet Captions dataset. Here, per unit length of input video represents non-overlapping clip of $2.5s$ duration for DiDeMo and non-overlapping clip of $1s$ duration for both Charades-STA and ActivityNet Captions dataset. For DiDeMo dataset, we use a fully connected layer followed by max-pool to generate representations with temporal dimension 6 for each video. Then we use 6 temporal convolutional layers to generate representations with temporal dimensions of $\{6, 5, 4, 3, 2, 1\}$ resulting in representations for 21 candidate moments. Similarly for Charades-STA, we use a fully connected layer followed by max-pool to generate representations with temporal dimension 32 for each video. Then we use 6 temporal convolutional layers with the temporal dimension of $\{32, 16, 8, 4, 2, 1\}$ where we use the 31 candidate moment representations from the last 5 layers. Additionally, we use a branch temporal convolutional layer to generate representations of 30 overlapping candidate moments, each with $6s$ duration and $2s$ stride. Combining these, we consider 61 candidate moments for each video of Charades-STA dataset. For ActivityNet Captions dataset, we use a feedforward network followed by 10 convolutional layers to generate representations with temporal dimension of $\{512, 256, 128, 64, 32, 16, 8, 4, 2, 1\}$, resulting in 1023 candidate moment representations. Table II illustrates the implementation details for video processing for all three datasets. we consider sentences with maximum of 15 words in length. If a sentence contains more than 15 words, the tailing words are truncated.

The proposed network is implemented in TensorFlow and trained using a single RTX 2080 GPU. To train the HMAN network, we use mini-batches containing 64 video-sentence pairs for DiDeMo and Charades-STA and 32 video-sentence pairs for ActivityNet Captions. We use the learning rate with exponential decay initializing from 10^{-3} for all three datasets. ADAM optimizer is used to train the network. We use 0.9 as the exponential decay rate for the first moment estimates and 0.999 as the exponential decay rate for the second-moment estimates. We set α_{intra} and α_{video} to 0.05 and 0.20, respectively for all three datasets. λ_1 is empirically set to 5, 1, and 1.5, respectively for DiDeMo, Charades-STA, and ActivityNet Captions. α is set to 5×10^{-5} for all three datasets.

D. Result Analysis

We conduct the following experiments to evaluate the performance of our proposed method:

- Comparison of the performance of proposed HMAN for the task of temporal localization of moments in video corpus with different baseline approaches and a concurrent work.
- Evaluation of the effectiveness of utilizing hierarchical moment encoder module.
- Investigation of the impact of learning joint embedding space by utilizing different components of the loss function (learning for intra-video moments ($\mathcal{L}^{intra}$) and learning for videos ($\mathcal{L}^{video}$).
- Evaluation of the effectiveness of utilizing global semantics to identify the correct video.
- Analyzing the effectiveness of video relevance computation (Eqn. 6) for the task of temporal localization of moments in a video corpus.
- Studying the performance of proposed HMAN for different visual features.
- Performance comparison of HMAN with decreasing number of test set moment-sentence pairs.
- Evaluation of the run time efficiency.
- Analysis of the λ_1 parameter sensitivity.

Temporal Localization of Moments in Video Corpus. Table III, Table IV, and Table V illustrate the quantitative perfor-

TABLE IX
COMPARISON OF THE PERFORMANCE OF PROPOSED LOGSUMEXP POOLING AND AVERAGE POOLING. WE COMPARE THE PERFORMANCE FOR THE TASK OF TEMPORAL LOCALIZATION OF MOMENTS IN VIDEO CORPUS FOR DIDEMO AND CHARADES-STA DATASET.

	DiDeMo				Charades-STA			
	$IoU = 0.50$		$IoU = 0.70$		$IoU = 0.50$		$IoU = 0.70$	
	R@10	R@100	R@10	R@100	R@10	R@100	R@10	R@100
HMAN (sum, ave)	5.63	26.05	4.43	20.82	1.10	7.19	0.62	**4.47**
HMAN (sum, log)	**6.25**	**28.39**	**4.98**	**22.51**	**1.29**	**7.73**	**0.83**	4.12
HMAN (max, ave)	5.27	17.65	**4.01**	13.60	0.75	7.00	0.51	4.53
HMAN (max, log)	**5.47**	**20.82**	3.86	**16.28**	**1.40**	**7.79**	**1.05**	**4.69**

TABLE X
ABLATION STUDY OF THE PERFORMANCE OF HMAN (SUM-MARGIN) FOR DIFFERENT VISUAL FEATURES FOR DIDEMO DATASET.

	$IoU = 0.50$		$IoU = 0.70$	
	R@10	R@100	R@10	R@100
VGGNet	2.61	16.36	1.79	12.82
VGGNet + Flow	3.98	21.29	3.14	16.76
ResNet	5.63	26.49	4.51	20.82
ResNet + Flow	**6.25**	**28.39**	**4.98**	**22.51**

TABLE XI
ABLATION STUDY OF THE PERFORMANCE OF HMAN (SUM-MARGIN) WHEN THE NUMBER OF TEST SET DATA IS DECREASED FOR DIDEMO DATASET.

	$IoU = 0.50$			$IoU = 0.70$		
	R@10	R@100	MR↓	R@10	R@100	MR↓
HMAN (100%)	6.25	28.39	302	4.98	22.51	416
HMAN (50%)	6.90	30.15	268	5.68	23.73	372
HMAN (25%)	8.74	34.93	193	7.06	27.62	269
HMAN (10%)	13.35	45.60	102	10.30	36.65	142

mance of our framework for the task of temporal localization of moments in the video corpus. The evaluation setup considers $IoU \in \{0.50, 0.70\}$ and for each IoU threshold, we report $R@10$, $R@100$ and MR. For a query sentence, the task requires to search over all the videos and retrieve the relevant moment. For example, in the DiDeMo dataset, the test set consists of $1{,}004$ videos totaling $4{,}016$ moment-sentence pairs. Again, we consider 21 candidate moments for each video. So, for each query sentence, we need to search over $21 \times 1{,}004 = 21{,}084$ moment instances and retrieve the correct moment. This is itself a difficult task and the addition of ambiguity of similar kinds of activities in different videos makes the problem even harder. We compare the proposed method with the following baselines:

- **Moment Frequency Prior:** We use Moment Frequency Prior baseline from [6], which selects moments that correspond to gifs most frequently described by the annotators.
- **MCN:** The Moment Context Network [6] for temporal localization of moments in a given video is scaled up to search moment from the entire video corpus.
- **SCDM:** The state-of-the-art Semantic Conditioned Dynamic Modulation (SCDM) network [13] for temporal localization of moments in a video is scaled up to search over the entire video corpus.
- **VSE++ + SCDM:** We use joint embedding based retrieval approach (VSE++ [63]) combined with SCDM as a baseline. In this setup, the framework first retrieves a few relevant videos (top 5%) and then localize moments on those retrieved videos using SCDM approach.
- **CAL:** We compare with Clip Alignment of Language [23]. It is a concurrent work that addresses the task of localizing moments in a video corpus by aligning clip representation with language representation in the embedding space.

Note that we do not compare with baselines that utilize temporal endpoint features from [6], as these directly correspond to dataset priors and do not reflect a model's capability [57].

We observe that MCN and CAL perform better than the state-of-the-art SCDM approach in DiDeMo dataset but perform poorly compared to the SCDM approach in Charades-STA dataset. This is due to the fact that the video contents and language queries differ a lot among different datasets [12]. MCN and CAL learn to distinguish both intra-video moments and inter-video moments locally while SCDM only learns to distinguish intra-video moments. As DiDeMo dataset contains diverse videos of different concepts and relatively less number of candidate moments, learning to differentiate inter-video moments locally improves performance significantly. However, learning to differentiate inter-video moments locally does not have much impact on Charades-STA dataset. This also indicates the importance of distinguishing moments from different videos based on global semantics for a diverse set of video datasets. We also observe that in some of the cases, VSE++ + SCDM scores drop compared to the SCDM approach. Since the performance of VSE++ + SCDM depends on retrieving correct video, the localization performance drops if the retrieval approach fails to retrieve correct videos with higher accuracy.

For HMAN, we report the performance for both sum-margin and max-margin based triplet loss setups. Additionally, for DiDeMo and Charades-STA dataset, we report the performance of HMAN for two different loss calculation setups: TripSiam [64] and DSLT [65]. In Table III, Compared to baseline approaches, the performance of our proposed approach is better for all metrics and outperforms other approaches with a maximum of 11.88% absolute improvement in DiDeMo dataset. We observe that the sum-margin based triplet loss setup outperforms the max-margin setup, while both of these setups perform better than other baselines in DiDeMo dataset. For a fair comparison with CAL and MCN, we report the performance of HMAN with the ResNet-152 feature computed from RGB frames only. This setup also outperforms CAL and

TABLE XII
PER EPOCH TRAINING AND INFERENCE TIME FOR CHARADES-STA DATASET.

Approach	Training time	Inference time
Sliding-based	35.05 s	90.46 s
HMAN	21.18 s	83.91 s

MCN. We also conduct experiment incorporating temporal end point feature in HMAN for DiDeMo dataset. It results in $\sim 0.5\% - 1\%$ improvement over HMAN (sum-margin) in $R@k$ metrics. It indicates the bias in the dataset where different types of events are correlated with different time frames of the video. In Table IV, for the Charades-STA dataset, the performance of HMAN is better for all metrics and the max-margin based triplet loss setup outperforms other baseline approaches with a maximum of 3.4% absolute improvement. In Table V, for ActivityNet Captions dataset, the HMAN max-margin setup outperforms other baselines with a maximum of 3.17% absolute improvement. We do not compute SCDM and VSE++ + SCDM baselines for ActivityNet Captions dataset. Moment representations in SCDM and VSE++ + SCDM approaches are conditioned on sentence queries. For each query sentence, we need to compute moment representations from all the videos, resulting in $O(n^2)$ complexity. So testing on a set of $34{,}160$ query sentences and $4{,}917 \times 1{,}023 = 5{,}030{,}091$ moment representations is impractical using these approaches.

TripSiam [64] and DSLT [65] are two different variants of triplet loss which are used in object tracking. TripSiam defines a matching probability for each triplet to measure the possibility of assigning the positive instance to exemplar compared with the negative instance and tries to maximize the joint probability among all triplets during training. DSLT [65] utilizes modulating function to minimize the contribution of easy samples in the total loss. While both setups perform better than baseline approaches, we observe that there is a significant improvement in median retrieval rank (MR). This indicates that even if TripSiam and DSLT can not retrieve the correct moment, they are robust in terms of the semantic association between moments and sentences.

Effectiveness of Hierarchical Moment Encoder. HMAN utilizes stacked temporal convolutional layers in a hierarchical structure to represent video moments. We conduct experiments to analyze the effects of using the hierarchical moment encoder module in our proposed model. We consider two setups, i) **w/ TCN**: the hierarchical moment encoder module built using temporal convolutional network is present in the model and ii) **w/o TCN**: the hierarchical moment encoder module is replaced with a simple feedforward network to project the candidate moment representations in the joint embedding space. We consider both sum-margin based and max-margin based triplet loss to train the networks. Table VI illustrates the effect of utilizing hierarchical moment encoder module. We observe that for both the learning approaches and for both datasets, there is a significant improvement in performance when the hierarchical moment encoder module is used. For example, in DiDeMo dataset, we observe $\sim 14\%$ (sum-margin) and $\sim 8\%$ (max-margin) absolute improvement in performance for $R@100, IoU = 0.50$.

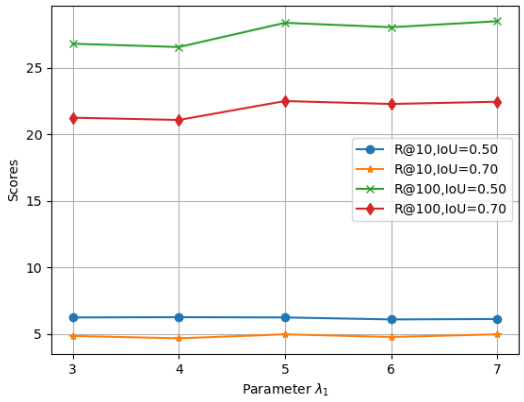

Fig. 4. Illustration of λ_1 parameter sensitivity on the HMAN performance. We observe that for the set of values $\{3, 4, 5, 6, 7\}$, performance of HMAN is stable.

Ablation Study of Learning Joint Embedding Space. We conduct experiments to analyze the impact of different components of the loss function to learn the joint embedding space for our targeted task in DiDeMo dataset and reported the results in Table VII. We use three setups to learn the joint embedding space:

- **HMAN (intra)**: Only uses $\mathcal{L}^{intra}$. So the network only learns to distinguish intra-video moments.
- **HMAN (video)**: Only uses $\mathcal{L}^{video}$. So the network only learns to disntinguish moments from different videos based on global semantics.
- **HMAN (proposed)**: Our proposed approach, combination of $\mathcal{L}^{intra}$ and $\mathcal{L}^{video}$.

In Table VII, we observe that the performance of HMAN is poor for both the case of HMAN (intra) and HMAN (video). Performance of HMAN (intra) is better compared to HMAN (video) in Table VII when higher IoU threshold requirement is considered ($R@k, IoU = 0.7$). This indicates that HMAN (intra) learns to better identify temporal boundaries in a video compared to HMAN (video), while HMAN (video) is better at distinguishing moments from different videos compared to HMAN (intra). However, when we combine both of these criteria, there is a significant performance boost as the model is able to effectively learn to identify both the correct video and the temporal boundary. All the results in Table VII are reported for sum-margin based triplet loss setup.

Effectiveness of Utilizing Global Semantics. Our proposed learning objective utilizes global semantics to distinguish moments from different videos. To do so, we learn to align corresponding video-sentence pairs, where the video-sentence relevance $R(v, s)$ in the embedding space is computed in terms of moment-sentence similarity $S(m, s)$. So we use this video-sentence relevance score $R(v, s)$ to analyse the models performance to identify or retrieve the correct video given a text query and report the results in Table VIII. We use the standard evaluation criteria $R@k$ for video retrieval task and report $R@10$, $R@100$, and $R@200$ scores for DiDeMo and

Charades-STA dataset. Here, $R@K$ calculates the percentage of query sentences for which the correct video is found in the top-K retrieved videos to the query sentence. In DiDeMo test set, there are $1{,}004$ videos with $4{,}016$ moment-sentence pairs (~ 4 sentences per video) and in Charades-STA testset, there are $1{,}334$ videos with $3{,}720$ moment-sentence pairs (~ 2.8 sentences per video). Due to the one-to-many correspondences, we consider $4{,}016$ and $3{,}720$ video-sentence pairs respectively for DiDeMo and Charades-STA datasets for the video retrieval task, where a single video can pair up with multiple sentences. Table VIII shows that both sum-margin (HMAN (sum)) and max-margin (HMAN (max)) based triplet loss setups of our proposed approach outperforms standard Visual Semantic Embedding based retrieval approach (VSE++) for the task of retrieving the correct video. Along with the consistent improvement of performance in all metrics for both datasets, We observe $\sim 40\%$ absolute improvement of retrieval performance for the metric R@200 for DiDeMo dataset. As the video-sentence relevance is computed in terms of moment-sentence similarity, this experiment validates the models capability to distinguish videos as well as moments from different videos utilizing global semantics.

Analysis of Video Relevance Computation Approach. In an untrimmed video with temporal language annotation, the segment/portion of the video mostly matches with the sentence semantics. So to compute the video-sentence relevance, it needs to focus on the moments that have higher similarity with the query sentence semantics. To tackle this issue, we compute the video-sentence relevance using LogSumExp pooling (Eqn. 6) of the moment-sentence similarity. In Table IX, we analyze the significance of the LogSumExp pooling compared to average pooling for both sum-margin and max margin based triplet loss setups. In Table IX, 'ave' and 'log' indicates average and LogSumExp pooling respectively, while 'sum' and 'max' indicates sum-margin based and max-margin based triplet loss respectively. For both DiDeMo and Charades-STA datasets, we observe that LogSumExp pooling performs better than average pooling for the target task of temporal localization of moments in video corpus in both sum-margin based and max-margin based triplet loss setups.

Ablation Study of Different Visual Features. We conduct experiments to study the performance of HMAN for different visual features for DiDeMo dataset and reported the results in Table X. We use extracted features from VGGNet [73], ResNet-152 [69] for RGB frames and optical flow features from [70]. In Table X, we observe that a combination of RGB and optical flow features perform better than using only an RGB stream. It indicates the models increased capacity due to the increase in the number of learnable weights. As a result, HMAN is suitable to work with multiple encodings of the same data together compared to the shallow embedding networks [6], [23]. We have reported the results for sum-margin based triplet loss setup.

Performance of HMAN on Decreased Number of Moment-sentence Pairs. Since HMAN searches for the correct candidate moment across all the videos in the test set during inference, the temporal localization performance of HMAN is expected to improve by decreasing the number of moment-sentence pairs in the test set. We conduct experiments on DiDeMo dataset to evaluate the performance of HMAN (learned using sum-margin based triplet loss) on the decreased number of moment-sentence pairs in the test phase. We consider four setups: **HMAN (100%)**: Model searches over the full test set during inference, **HMAN (50%)**: Model searches over each 50% of the test set separately and take the average of the scores, **HMAN (25%)**: Model searches over each 25% of test set separately and take the average of the scores, **HMAN (10%)**: Model searches over each 10% of test set separately and take the average of the scores. Table XI illustrates the performance for all four setups. We observe that with decreased number of test set moment-sentence pairs, the performance of HMAN improves.

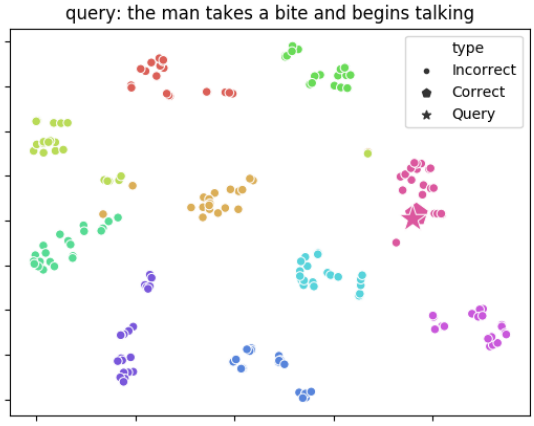

Fig. 5. t-SNE visualization of text query representation and candidate moment representations. Different color represents different video. The color of the text representation is the same as the corresponding video. We use different markers for the representation of incorrect candidate moments, correct candidate moments and text. Here, representations of the text query and the correct candidate moment coincide. Also, the representations of candidate moments from the same video are clustered together.

Evaluation of Run Time Efficiency. We conduct experiments on the Charades-STA dataset to compare the run time of HMAN with the sliding window-based approaches. The differences in the sliding-based approach compared to the setup of HMAN is that: i) the moment encoder module with temporal convolutional network of HMAN is replaced by a simple single layer feedforward network, ii) instead of generating candidate moment representations directly from the video, we slide over the video to extract features of different temporal durations, then use extracted features to generate candidate moment representations. Table XII illustrates that for both training case and inference case, the sliding-based approach takes longer than HMAN per epoch, even though the network is much smaller in the sliding-based approach compared to HMAN. For a fair comparison, we keep the number of candidate moments the same, and similar computations (apart from hierarchical moment encoder module replaced by single layer feed forward network) are done for both the approaches. We have computed the run time for five epochs and reported the average results. Here, the inference time is higher due to the added requirement of computing the

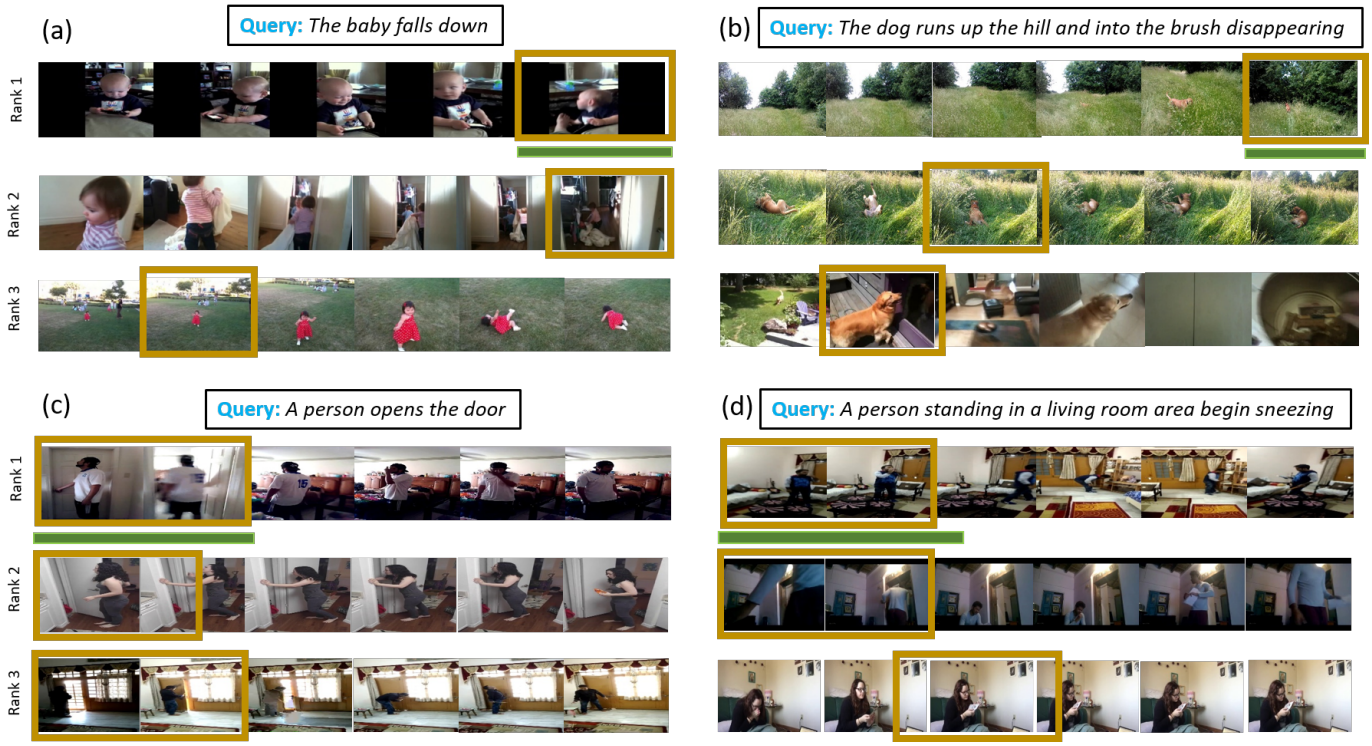

Fig. 6. Example illustration of the performance of HMAN for the task of localization of moments in a corpus of videos. For each query sentence, we display the top-3 retrieved moments. The retrieved moments are surrounded by gold boxes and the ground truth moments are indicated by green lines. We observe that for each of the queries, the top-3 retrieved moments are semantically related to the sentence proving the efficacy of our approach.

cosine distance between each text query and all the candidate moment representations.

λ_1 Parameter Sensitivity Analysis. In our framework, λ_1 balances the contribution of $\mathcal{L}^{intra}$ and $\mathcal{L}^{video}$ for both sum-margin and max-margin case. We choose the value of λ_1 empirically. We conduct an experiment to check the sensitivity of HMAN performance based on a set of values for λ_1 in the DiDeMo dataset where $\lambda_1 \in \{3, 4, 5, 6, 7\}$. In Figure 4 shows that for this set of values of λ_1, the performance is stable.

E. Qualitative Results

t-SNE Visualization. We provide t-SNE visualization of embedding representations of text query and candidate moments in Figure 5. For a text query, we consider embedding representation of the text query, representations of candidate moments from the correct video, and representations of candidate moments from randomly picked 9 other videos and visualize the distribution of representations. In Figure 5, different color represents different videos. Each video has 21 candidate moments. We keep the color of the text query representation the same as the color of candidate moments representation from the correct video and use separate markers for correct candidate moment and text query representation. We observe that representations of the text query and the correct candidate moment coincide. Also, the representations of candidate moments from the same video are clustered together.

Example Illustration. In Figure 6, we illustrate some qualitative results for our proposed approach. The two examples in the top row are for the DiDeMo dataset and the two examples in the bottom row are for the Charades-STA dataset. For each query sentence, we demonstrate the examples where the network is able to retrieve the correct moment as the rank-1 from the test set videos. We also display rank-2 and rank-3 moments retrieved by the model for each query sentence. Figure 6(a) shows that for the query *'The baby falls down'*, the model was able to retrieve the correct moment with the highest matching. However, the interesting fact lies in the retrieved rank-2 and rank-3 moments. For the query *'The baby falls down'*, the retrieved rank-2 and rank-3 moments also contain activity of a baby, including a baby falling down. Similar results are observed for other examples for both datasets. For example, in Figure 6(b), for the query sentence *'A person opens the door'*, the model was able to retrieve the correct moment with the highest matching. However, all top-3 ranked moments contain activity related to a door. In the rank-2 moment, a person is opening a door and in the rank-3 moment, a person is fixing a door. Similarly, the top retrieved moments for a query of a dog running and hiding contain activities of a dog (Figure 6(b)) and top retrieved moments for a query of a person standing and sneezing contain standing activity and sneezing activity (Figure 6(d)). These results indicate the model's capability of retrieving moments with similar semantic concepts from the corpus of videos.

V. Conclusion

In this work, we explore an important and under-explored task of localizing moments in a video corpus based on text query. We adapt existing temporal localization of moments

approaches and video retrieval approaches for the proposed task and identified the shortcomings of those approaches. Towards addressing the challenging task, we propose Hierarchical Moment Alignment Network (HMAN), a novel neural network that effectively learns a joint embedding space for video moments and sentences to retrieve the matching moment based on semantic closeness in the embedding space. Our proposed learning objective allows the model to identify subtle changes of intra-video moments as well as distinguish inter-video moments utilizing text-guided global semantic concepts of videos. We adopt both sum-margin based and max-margin based triplet loss setups separately and achieve performance improvement over other baseline approaches in both setups. We experimentally validate the effectiveness of our proposed approach on the DiDeMo, Charades-STA, and ActivityNet Captions datasets.

ACKNOWLEDGMENT

The work was partially supported by NSF grant IIS-1901379 and ONR grant N00014-19-1-2264.

REFERENCES

[1] P. Lei and S. Todorovic, "Temporal deformable residual networks for action segmentation in videos," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 6742–6751.

[2] Y.-W. Chao, S. Vijayanarasimhan, B. Seybold, D. A. Ross, J. Deng, and R. Sukthankar, "Rethinking the faster r-cnn architecture for temporal action localization," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 1130–1139.

[3] B. Singh, T. K. Marks, M. Jones, O. Tuzel, and M. Shao, "A multi-stream bi-directional recurrent neural network for fine-grained action detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 1961–1970.

[4] T. Lin, X. Zhao, and Z. Shou, "Single shot temporal action detection," in *Proceedings of the 25th ACM international conference on Multimedia*, 2017, pp. 988–996.

[5] J. Gao, C. Sun, Z. Yang, and R. Nevatia, "Tall: Temporal activity localization via language query," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 5267–5275.

[6] L. Anne Hendricks, O. Wang, E. Shechtman, J. Sivic, T. Darrell, and B. Russell, "Localizing moments in video with natural language," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 5803–5812.

[7] A. Wu and Y. Han, "Multi-modal circulant fusion for video-to-language and backward," in *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, 2018, pp. 1029–1035.

[8] M. Liu, X. Wang, L. Nie, X. He, B. Chen, and T.-S. Chua, "Attentive moment retrieval in videos," in *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, 2018, pp. 15–24.

[9] J. Chen, X. Chen, L. Ma, Z. Jie, and T.-S. Chua, "Temporally grounding natural sentence in video," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 2018, pp. 162–171.

[10] R. Ge, J. Gao, K. Chen, and R. Nevatia, "Mac: Mining activity concepts for language-based temporal localization," in *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE, 2019, pp. 245–253.

[11] H. Xu, K. He, B. A. Plummer, L. Sigal, S. Sclaroff, and K. Saenko, "Multilevel language and vision integration for text-to-clip retrieval," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, 2019, pp. 9062–9069.

[12] D. Zhang, X. Dai, X. Wang, Y.-F. Wang, and L. S. Davis, "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*, 2019, pp. 1247–1257.

[13] Y. Yuan, L. Ma, J. Wang, W. Liu, and W. Zhu, "Semantic conditioned dynamic modulation for temporal sentence grounding in videos," in *Advances in Neural Information Processing Systems*, 2019, pp. 534–544.

[14] Z. Lin, Z. Zhao, Z. Zhang, Z. Zhang, and D. Cai, "Moment retrieval via cross-modal interaction networks with query reconstruction," *IEEE Transactions on Image Processing*, vol. 29, pp. 3750–3762, 2020.

[15] B. Zhang, H. Hu, and F. Sha, "Cross-modal and hierarchical modeling of video and text," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 374–390.

[16] N. C. Mithun, J. Li, F. Metze, and A. K. Roy-Chowdhury, "Joint embeddings with multimodal cues for video-text retrieval," *International Journal of Multimedia Information Retrieval*, pp. 1–16, 2019.

[17] D. Shao, Y. Xiong, Y. Zhao, Q. Huang, Y. Qiao, and D. Lin, "Find and focus: Retrieve and localize video events with natural language queries," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 200–216.

[18] M. Wray, D. Larlus, G. Csurka, and D. Damen, "Fine-grained action retrieval through multiple parts-of-speech embeddings," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 450–459.

[19] J. Dong, X. Li, C. Xu, S. Ji, Y. He, G. Yang, and X. Wang, "Dual encoding for zero-example video retrieval," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 9346–9355.

[20] M. Qi, J. Qin, Y. Yang, Y. Wang, and J. Luo, "Semantics-aware spatial-temporal binaries for cross-modal video retrieval," *IEEE Transactions on Image Processing*, vol. 30, pp. 2989–3004, 2021.

[21] G. Wu, J. Han, Y. Guo, L. Liu, G. Ding, Q. Ni, and L. Shao, "Unsupervised deep video hashing via balanced code for large-scale video retrieval," *IEEE Transactions on Image Processing*, vol. 28, no. 4, pp. 1993–2007, 2018.

[22] Z. Feng, Z. Zeng, C. Guo, and Z. Li, "Exploiting visual semantic reasoning for video-text retrieval," *arXiv preprint arXiv:2006.08889*, 2020.

[23] V. Escorcia, M. Soldan, J. Sivic, B. Ghanem, and B. Russell, "Temporal localization of moments in video collections with natural language," *arXiv preprint arXiv:1907.12763*, 2019.

[24] Y. Pan, T. Mei, T. Yao, H. Li, and Y. Rui, "Jointly modeling embedding and translation to bridge video and language," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2016, pp. 4594–4602.

[25] M. Ye, J. Shen, X. Zhang, P. C. Yuen, and S.-F. Chang, "Augmentation invariant and instance spreading feature for softmax embedding," *IEEE transactions on pattern analysis and machine intelligence*, 2020.

[26] M. Ye and J. Shen, "Probabilistic structural latent representation for unsupervised embedding," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 5457–5466.

[27] K.-H. Lee, X. Chen, G. Hua, H. Hu, and X. He, "Stacked cross attention for image-text matching," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 201–216.

[28] L. Liu, Z. Lin, L. Shao, F. Shen, G. Ding, and J. Han, "Sequential discrete hashing for scalable cross-modality similarity retrieval," *IEEE Transactions on Image Processing*, vol. 26, no. 1, pp. 107–118, 2016.

[29] C. Deng, Z. Chen, X. Liu, X. Gao, and D. Tao, "Triplet-based deep hashing network for cross-modal retrieval," *IEEE Transactions on Image Processing*, vol. 27, no. 8, pp. 3893–3903, 2018.

[30] J. Dong, X. Li, and C. G. Snoek, "Predicting visual features from text for image and video caption retrieval," *IEEE Transactions on Multimedia*, vol. 20, no. 12, pp. 3377–3388, 2018.

[31] N. C. Mithun, J. Li, F. Metze, and A. K. Roy-Chowdhury, "Learning joint embedding with multimodal cues for cross-modal video-text retrieval," in *ACM International Conference on Multimedia Retrieval (ICMR)*. ACM, 2018.

[32] J. Xu, T. Mei, T. Yao, and Y. Rui, "Msr-vtt: A large video description dataset for bridging video and language," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 5288–5296.

[33] A. Rohrbach, M. Rohrbach, N. Tandon, and B. Schiele, "A dataset for movie description," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3202–3212.

[34] D. L. Chen and W. B. Dolan, "Collecting highly parallel data for paraphrase evaluation," in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*. Association for Computational Linguistics, 2011, pp. 190–200.

[35] F. Markatopoulou, D. Galanopoulos, V. Mezaris, and I. Patras, "Query and keyframe representations for ad-hoc video search," in *Proceedings of the 2017 ACM on International Conference on Multimedia Retrieval*, 2017, pp. 407–411.

[36] D.-D. Le, S. Phan, V.-T. Nguyen, B. Renoust, T. A. Nguyen, V.-N. Hoang, T. D. Ngo, M.-T. Tran, Y. Watanabe, M. Klinkigt *et al.*, "Nii-hitachi-uit at trecvid 2016," 2016.

[37] K. Ueki, "Waseda meisei at trecvid 2017: Ad-hoc video search," 2017.

[38] R. Xu, C. Xiong, W. Chen, and J. J. Corso, "Jointly modeling deep video and compositional text to bridge vision and language in a unified framework." in *AAAI*, vol. 5, 2015, p. 6.

[39] Y. Yu, J. Kim, and G. Kim, "A joint sequence fusion model for video question answering and retrieval," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 471–487.

[40] X. Li, C. Xu, G. Yang, Z. Chen, and J. Dong, "W2vv++ fully deep learning for ad-hoc video search," in *Proceedings of the 27th ACM International Conference on Multimedia*, 2019, pp. 1786–1794.

[41] Z. Chen, L. Ma, W. Luo, and K.-Y. K. Wong, "Weakly-supervised spatio-temporally grounding natural sentence in video," *arXiv preprint arXiv:1906.02549*, 2019.

[42] Y. Yu, H. Ko, J. Choi, and G. Kim, "End-to-end concept word detection for video captioning, retrieval, and question answering," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 3165–3173.

[43] Y. Liu, S. Albanie, A. Nagrani, and A. Zisserman, "Use what you have: Video retrieval using representations from collaborative experts," *arXiv preprint arXiv:1907.13487*, 2019.

[44] Y. Song and M. Soleymani, "Polysemous visual-semantic embedding for cross-modal retrieval," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 1979–1988.

[45] S. Chen, Y. Zhao, Q. Jin, and Q. Wu, "Fine-grained video-text retrieval with hierarchical graph reasoning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 10 638–10 647.

[46] B. Jiang, X. Huang, C. Yang, and J. Yuan, "Cross-modal video moment retrieval with spatial and language-temporal attention," in *Proceedings of the 2019 on International Conference on Multimedia Retrieval*, 2019, pp. 217–225.

[47] M. Liu, X. Wang, L. Nie, Q. Tian, B. Chen, and T.-S. Chua, "Cross-modal moment localization in videos," in *Proceedings of the 26th ACM international conference on Multimedia*, 2018, pp. 843–851.

[48] S. Zhang, J. Su, and J. Luo, "Exploiting temporal relationships in video moment localization with natural language," in *Proceedings of the 27th ACM International Conference on Multimedia*, 2019, pp. 1230–1238.

[49] N. C. Mithun, S. Paul, and A. K. Roy-Chowdhury, "Weakly supervised video moment retrieval from text queries," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[50] S. Ghosh, A. Agarwal, Z. Parekh, and A. Hauptmann, "Excl: Extractive clip localization using natural language descriptions," *arXiv preprint arXiv:1904.02755*, 2019.

[51] Y. Yuan, T. Mei, and W. Zhu, "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, 2019, pp. 9159–9166.

[52] Z. Zhang, Z. Lin, Z. Zhao, and Z. Xiao, "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*, 2019, pp. 655–664.

[53] S. Zhang, H. Peng, J. Fu, and J. Luo, "Learning 2d temporal adjacent networks for moment localization with natural language," *arXiv preprint arXiv:1912.03590*, 2019.

[54] D. He, X. Zhao, J. Huang, F. Li, X. Liu, and S. Wen, "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, 2019, pp. 8393–8400.

[55] M. Hahn, A. Kadav, J. M. Rehg, and H. P. Graf, "Tripping through time: Efficient localization of activities in videos," *arXiv preprint arXiv:1904.09936*, 2019.

[56] L. A. Hendricks, O. Wang, E. Shechtman, J. Sivic, T. Darrell, and B. Russell, "Localizing moments in video with temporal language," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 2018, pp. 1380–1390.

[57] B. Liu, S. Yeung, E. Chou, D.-A. Huang, L. Fei-Fei, and J. Carlos Niebles, "Temporal modular networks for retrieving complex compositional activities in videos," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 552–568.

[58] M. Regneri, M. Rohrbach, D. Wetzel, S. Thater, B. Schiele, and M. Pinkal, "Grounding action descriptions in videos," *Transactions of the Association for Computational Linguistics*, vol. 1, pp. 25–36, 2013.

[59] A. Frome, G. S. Corrado, J. Shlens, S. Bengio, J. Dean, T. Mikolov *et al.*, "Devise: A deep visual-semantic embedding model," in *Advances in Neural Information Processing Systems (NIPS)*, 2013, pp. 2121–2129.

[60] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 2014, pp. 1532–1543.

[61] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," *arXiv preprint arXiv:1412.3555*, 2014.

[62] C. Lea, R. Vidal, A. Reiter, and G. D. Hager, "Temporal convolutional networks: A unified approach to action segmentation," in *European Conference on Computer Vision*. Springer, 2016, pp. 47–54.

[63] F. Faghri, D. J. Fleet, J. R. Kiros, and S. Fidler, "Vse++: Improving visual-semantic embeddings with hard negatives," *arXiv preprint arXiv:1707.05612*, 2017.

[64] X. Dong and J. Shen, "Triplet loss in siamese network for object tracking," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 459–474.

[65] X. Lu, C. Ma, J. Shen, X. Yang, I. Reid, and M.-H. Yang, "Deep object tracking with shrinkage loss," *IEEE transactions on pattern analysis and machine intelligence*, 2020.

[66] R. Krishna, K. Hata, F. Ren, L. Fei-Fei, and J. C. Niebles, "Dense-captioning events in videos," in *International Conference on Computer Vision (ICCV)*, 2017.

[67] G. A. Sigurdsson, G. Varol, X. Wang, A. Farhadi, I. Laptev, and A. Gupta, "Hollywood in homes: Crowdsourcing data collection for activity understanding," in *European Conference on Computer Vision*. Springer, 2016, pp. 510–526.

[68] F. C. Heilbron, V. Escorcia, B. Ghanem, and J. C. Niebles, "Activitynet: A large-scale video benchmark for human activity understanding," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2015, pp. 961–970.

[69] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.

[70] L. Wang, Y. Xiong, Z. Wang, Y. Qiao, D. Lin, X. Tang, and L. Van Gool, "Temporal segment networks: Towards good practices for deep action recognition," in *European Conference on Computer Vision (ECCV)*. Springer, 2016, pp. 20–36.

[71] J. Carreira and A. Zisserman, "Quo vadis, action recognition? a new model and the kinetics dataset," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2017, pp. 4724–4733.

[72] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3d convolutional networks," in *International Conference on Computer Vision (ICCV)*. IEEE, 2015, pp. 4489–4497.

[73] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

Sudipta Paul received his Bachelor's degree in Electrical and Electronic Engineering from Bangladesh University of Engineering and Technology, Dhaka in 2016. He is currently pursuing his Ph.D. degree in the department of Electrical and Computer Engineering at University of California, Riverside. His main research interests include computer vision, machine learning, vision and language, and robust learning.

Niluthpol Chowdhury Mithun graduated from the University of California, Riverside with a Ph.D. in Electrical and Computer Engineering in 2019. Previously, he received his Bachelors and Masters degree from Bangladesh University of Engineering and Technology. He is currently an Advanced Computer Scientist at the Center for Vision Technologies, SRI International, Princeton. His broad research interest includes computer vision and machine learning with more focus on multimodal data analysis, weakly supervised learning and vision-based localization.

Amit Roy-Chowdhury received his PhD from the University of Maryland, College Park (UMCP) in 2002 and joined the University of California, Riverside (UCR) in 2004 where he is a Professor and Bourns Family Faculty Fellow of Electrical and Computer Engineering, Director of the Center for Robotics and Intelligent Systems, and Cooperating Faculty in the department of Computer Science and Engineering. He leads the Video Computing Group at UCR, working on foundational principles of computer vision, image processing, and statistical learning, with applications in cyber-physical, autonomous and intelligent systems. He has published over 200 papers in peer-reviewed journals and conferences. He has published two monographs: Camera Networks: The Acquisition and Analysis of Videos Over Wide Areas and Person Re-identification with Limited Supervision. He is on the editorial boards of major journals and program committees of the main conferences in his area. His students have been first authors on multiple papers that received Best Paper Awards at major international conferences. He is a Fellow of the IEEE and IAPR, received the Doctoral Dissertation Advising/Mentoring Award 2019 from UCR, and the ECE Distinguished Alumni Award from UMCP.