

# Cross-Modal and Hierarchical Modeling of Video and Text

Bowen Zhang<sup>\*1</sup>, Hexiang Hu<sup>\*1</sup>, and Fei Sha<sup>2</sup>

<sup>1</sup> Dept. of Computer Science, U. of Southern California, Los Angeles, CA 90089

<sup>2</sup> Netflix, 5808 Sunset Blvd, Los Angeles, CA 90028

[zhan734@usc.edu](mailto:zhan734@usc.edu), [hexiangh@usc.edu](mailto:hexiangh@usc.edu), [fsha@netflix.com](mailto:fsha@netflix.com)\*\*

**Abstract.** Visual data and text data are composed of information at multiple granularities. A video can describe a complex scene that is composed of multiple clips or shots, where each depicts a semantically coherent event or action. Similarly, a paragraph may contain sentences with different topics, which collectively conveys a coherent message or story. In this paper, we investigate the modeling techniques for such hierarchical sequential data where there are correspondences across multiple modalities. Specifically, we introduce hierarchical sequence embedding (HSE), a generic model for embedding sequential data of different modalities into hierarchically semantic spaces, with either explicit or implicit correspondence information. We perform empirical studies on large-scale video and paragraph retrieval datasets and demonstrated superior performance by the proposed methods. Furthermore, we examine the effectiveness of our learned embeddings when applied to downstream tasks. We show its utility in zero-shot action recognition and video captioning.

**Keywords:** Hierarchical Sequence Embedding, Video Text Retrieval, Video Description Generation, Action Recognition, Zero-shot Transfer

## 1 Introduction

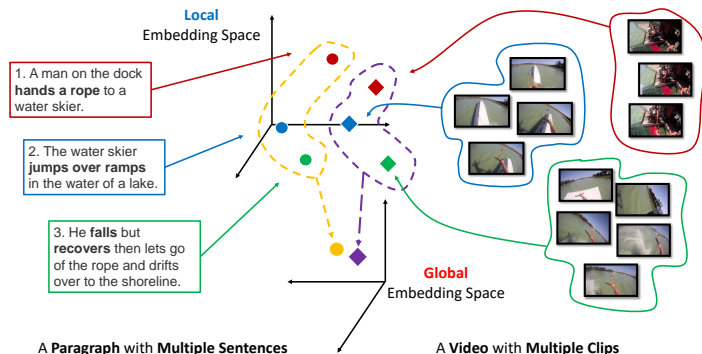
Recently, there has been an intensive interest in multi-modal learning of vision + language. A few challenging tasks have been proposed: visual semantic embedding (VSE) [16, 15, 5], image captioning [37, 42, 12, 21], and visual question answering (VQA) [2, 47, 3]. To jointly understand these two modalities of data and make inference over them, the main intuition is that different types of data can share a common semantic representation space. Examples are embedding images and the visual categories [7], embedding images and texts for VSE [16], and embedding images, questions, and answers for VQA [11]. Once embedded into this common (vector) space, similarity and distances among originally heterogeneous data can be captured by learning algorithms.

While there has been a rich study on how to discover this shared semantic representation on structures such as images, noun phrases (visual object or

---

\* equal contribution

\*\* On leave from U. of Southern California ([feisha@usc.edu](mailto:feisha@usc.edu))



**Fig. 1.** Conceptual diagram of our approach for cross-modal modeling of video and texts. The main idea is to embed both low-level (clips and sentences) and high-level (video and paragraph) in their own semantic spaces coherently. As shown in the figure, the 3 sentences (and the corresponding 3 clips) are mapped into a local embedding space where the corresponding pairs of clips and sentences are placed close to each other. As a whole, the videos and the paragraphs are mapped into a global semantic space where their embeddings are close. See Fig. 3 and texts for details.

action categories) and sentences (such as captions, questions, answers), less is known about how to achieve so on more complex structures such as videos and paragraphs of texts<sup>3</sup>. There are conceptual challenges: while complex structured data can be mapped to vector spaces (for instance, using deep architectures [18, 8]), it is not clear whether the intrinsic structures in those data’s original format, after being transformed to the vectorial representations, still maintain their correspondence and relevance across modalities.

Take the dense video description task as an example [17]. The task is to describe a video which is made of short, coherent and meaningful clips. (Note that those clips could overlap temporally.) Due to its narrowly focused semantic content, each clip is then describable with a sentence. The description for the whole video is then a paragraph of texts with sentences linearly arranged in order. Arguably, a corresponding pair of video and its descriptive paragraph can be embedded into a semantic space where their embeddings are close to each other, using a vanilla learning model by ignoring the boundaries of clips and sentences and treating as a sequence of continually flowing visual frames and words. However, for such a modeling strategy, it is opaque that if and how the correspondences at the “lower level” (*i.e.* clips versus sentences) are useful in either deriving the embeddings or using the embeddings to perform downstream tasks such as video or text retrieval.

Addressing these deficiencies, we propose a novel cross-modal learning approach to model both videos and texts jointly. The main idea is schematically illustrated in Fig. 1. Our approach is mindful of the intrinsic hierarchical struc-

<sup>3</sup> We use paragraphs and documents interchangeably throughout this work.

tures of both videos and texts, and models them with hierarchical sequence learning models such as GRUs [4]. However, as opposed to methods which disregard low-level correspondences, we exploit them by deriving loss functions to ensure the embeddings for the clips and sentences are also in accordance in their own (shared) semantic space. Those low-level embeddings in turn strengthen the desiderata that videos and paragraphs are embedded coherently. We demonstrate the advantages of the proposed model in a range of tasks including video and text retrieval, zero-shot action recognition and video description.

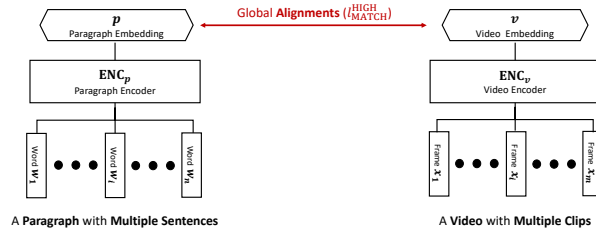
The rest of the paper is organized as follows. In section 2, we discuss related work. We describe our proposed approach in section 3, followed by extensive experimental results and ablation studies in section 4. We conclude in section 5.

## 2 Related Work

*Hierarchical Sequence Embedding Models.* Embedding images, videos, and textual data has been very popular with the rise of deep learning. The most related works to ours are [19] and [25]. The former models the paragraph using a hierarchical auto-encoder for text modeling [19], and the later uses a hierarchical RNN for videos and a one-layer RNN for caption generation. In contrast, our work models both modalities hierarchically and learn the parameters by leveraging the correspondences across modalities. Works motivated by other application scenarios usually explore hierarchical modeling in one modality [24, 43, 45].

*Cross-modal Embedding Learning.* There has been a rich history to learn embeddings for images and smaller linguistic units (such as words and noun phrases). DeVISE [7] learns to align the latent embeddings of visual data and names of the visual object categories. ReViSE [34] uses auto-encoders to derive embeddings for images and words which allow them to leverage unlabeled data. In contrast to previous methods, our approach models both videos and texts hierarchically, bridging the embeddings at different granularities using discriminative loss computed on corresponded pairs (*i.e.* videos vs. paragraphs).

*Action Recognition in Videos.* Deep learning has brought significant improvement to video understanding [30, 33, 6, 38, 44, 41] on large-scale action recognition datasets [9, 31, 14] in the past decade. Most of them [30, 6, 38] employed deep convolutional neural network to learn appearance feature and motion information respectively. Based on the spatial-temporal feature from these video modeling methods, we learn video semantic embedding to match the holistic video representation to text representation. To evaluate the generalization of our learned video semantic representation, we evaluate the model directly on the challenging action recognition benchmark. (Details in Section 4.4)



**Fig. 2.** Flat sequence modeling of videos and texts, ignoring the hierarchical structures in either and regarding the video (paragraph) as a sequence of frames (words).

### 3 Approach

We begin by describing the problem settings and introducing necessary notations. We then describe the standard sequential modeling technique, ignoring the hierarchical structures in the data. Finally, we describe our approach.

#### 3.1 Settings and Notations

We are interested in modeling videos and texts that are paired in correspondence. In the later section, we describe how to generalize this where there is no one to one correspondence.

A video  $v$  has  $n$  clips (or subshots), where each clip  $c_i$  contains  $n_i$  frames. Each frame is represented by a visual feature vector  $\mathbf{x}_{ij}$ . This feature vector can be derived in many ways, for instance, by feeding the frame (and its contextual frames) to a convolution neural net and using the outputs from the penultimate layer. Likewise, we assume there is a paragraph of texts describing the video. The paragraph  $p$  contains  $n$  sentences, one for each video clip. Let  $s_i$  denote the  $i$ th sentence and  $\mathbf{w}_{ij}$  the feature for the  $j$ th word out of  $n'_i$  words. We denote by  $\mathcal{D} = \{(v_k, p_k)\}$  a set of corresponding videos and text descriptions.

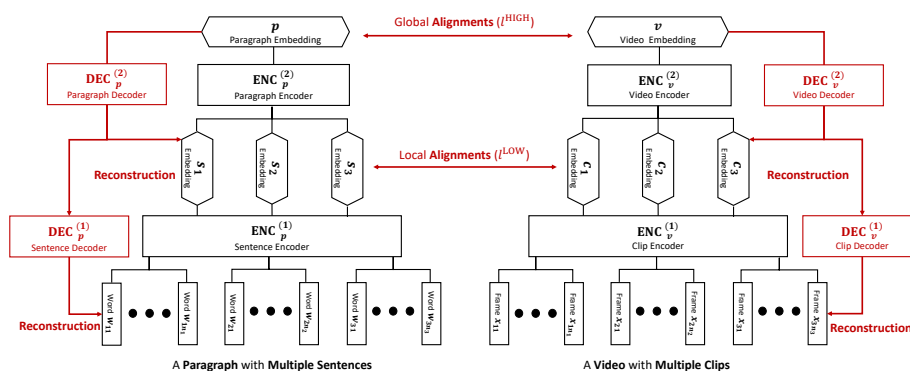
We compute a clip vector embedding  $\mathbf{c}_i$  from the frame features  $\{\mathbf{x}_{ij}\}$ , and a sentence embedding  $\mathbf{s}_i$  from the word features  $\{\mathbf{w}_{ij}\}$ . From those, we derive  $\mathbf{v}$  and  $\mathbf{p}$ , the embedding for the video and the paragraph, respectively.

#### 3.2 Flat Sequence Modeling

Many sequence-to-sequence (SEQ2SEQ) methods leverage the encoder-decoder structure [32, 22] to model the process of transforming from the input sequence to the output sequence. In particular, the encoder, which is composed of a layer of long short-term memory units (LSTMs) [10] or Gated Recurrent Units (GRUs) [4], transforms the input sequence into a vector as the embedding  $\mathbf{h}$ . The similarly constructed decoder takes  $\mathbf{h}$  as input and outputs another sequence.

The original SEQ2SEQ methods do not consider the hierarchical structures in videos or texts. We refer the embeddings as *flat sequence embedding* (FSE):

$$\mathbf{v} = \text{ENC}_v(\{\mathbf{x}_{ij}\}), \quad \mathbf{p} = \text{ENC}_p(\{\mathbf{w}_{ij}\}), \quad (1)$$



**Fig. 3.** Hierarchical cross-modal modeling of videos and texts. We differ from previous works [19, 25] in two aspects (components in red color): layer-wise reconstruction through decoders, and matching at both global and local levels. See texts for details.

Fig. 2 schematically illustrates this idea. We measure how well the videos and the texts are aligned by the following cosine similarity

$$\text{MATCH}(v, p) = \mathbf{v}^\top \mathbf{p} / \|\mathbf{v}\| \|\mathbf{p}\| \quad (2)$$

### 3.3 Hierarchical Sequence Modeling

One drawback of flat sequential modeling is that the LSTM/GRU layer needs to have a sufficient number of units to model well the potential long-range dependency among video frames (or words). This often complicates learning as the optimization becomes difficult [26].

We leverage the hierarchical structures in those data to overcome this deficiency: a video is made of clips which are made of frames. In parallel, a paragraph of texts is made of sentences which in turn are made of words. Similar ideas have been explored in [25, 19] and other previous works. The basic idea is illustrated in Fig. 3, where we also add components in red color to highlight our extensions.

**Hierarchical Sequence Embedding.** Given the hierarchical structures in Fig. 3, we can compute the embeddings using the forward paths

$$\begin{aligned} \mathbf{c}_i &= \text{ENC}_v^{(1)}(\{\mathbf{x}_{ij}, j = 1, 2, n_i\}), \quad \mathbf{v} = \text{ENC}_v^{(2)}(\{\mathbf{c}_i\}) \\ \mathbf{s}_i &= \text{ENC}_p^{(1)}(\{\mathbf{w}_{ij}, j = 1, 2, n'_i\}), \quad \mathbf{p} = \text{ENC}_p^{(2)}(\{\mathbf{s}_i\}) \end{aligned} \quad (3)$$

**Learning with Discriminative Loss.** For videos and texts have strong correspondences where clips and sentences are paired, we optimize the encoders such that videos and texts are matched. To this end, we define two loss functions,

corresponding to the matching at the low-level and the high-level respectively:

$$\begin{aligned} \ell_{\text{MATCH}}^{\text{HIGH}} &= \sum_k \sum_{k' \neq k} [\alpha + \text{MATCH}(\mathbf{v}_k, \mathbf{p}_k) - \text{MATCH}(\mathbf{v}_{k'}, \mathbf{p}_k)]_+ \\ &\quad + [\alpha + \text{MATCH}(\mathbf{v}_k, \mathbf{p}_k) - \text{MATCH}(\mathbf{v}_k, \mathbf{p}_{k'})]_+ \quad (4) \\ \ell_{\text{MATCH}}^{\text{LOW}} &= \sum_k \sum_i \sum_{(k', i') \neq (k, i)} [\beta + \text{MATCH}(\mathbf{c}_{ki}, \mathbf{s}_{ki}) - \text{MATCH}(\mathbf{c}_{k'i'}, \mathbf{s}_{ki})]_+ \\ &\quad + [\beta + \text{MATCH}(\mathbf{c}_{ki}, \mathbf{s}_{ki}) - \text{MATCH}(\mathbf{c}_{ki}, \mathbf{s}_{k'i'})]_+ \quad (5) \end{aligned}$$

These losses are margin-based losses [29] where  $\alpha$  and  $\beta$  are positive numbers as the margins to separate matched pairs from unmatched ones. The function  $[\cdot]_+$  is the standard hinge loss function.

**Learning with Contrastive Loss.** Assuming videos and texts are well clustered, we use the following loss to model their clustering in their own space.

$$\begin{aligned} \ell_{\text{CLUSTER}}^{\text{HIGH}} &= \sum_k \sum_{k' \neq k} [\gamma + 1 - \text{MATCH}(\mathbf{v}_{k'}, \mathbf{v}_k)]_+ + [\gamma + 1 - \text{MATCH}(\mathbf{p}_{k'}, \mathbf{p}_k)]_+ \quad (6) \\ \ell_{\text{CLUSTER}}^{\text{LOW}} &= \sum_k \sum_i \sum_{(k', i') \neq (k, i)} [\eta + 1 - \text{MATCH}(\mathbf{c}_{k'i'}, \mathbf{c}_{ki})]_+ \\ &\quad + [\eta + 1 - \text{MATCH}(\mathbf{s}_{k'i'}, \mathbf{s}_{ki})]_+ \quad (7) \end{aligned}$$

Note that the self-matching values  $\text{MATCH}(\mathbf{v}_k, \mathbf{v}_k)$  and  $\text{MATCH}(\mathbf{p}_k, \mathbf{p}_k)$  are 1 by definition. This loss can be computed on videos and texts alone and does not require them being matched.

**Learning with Unsupervised Layer-wise Reconstruction Loss.** Thus far, the matching loss focuses on matching across modality. The clustering loss focuses on separating between video/text data so that they do not overlap. None of them, however, focuses on the *quality* of the modeling data itself. In what follows, we propose a layer-wise reconstruction loss – when minimized, this loss ensures the learned video/text embedding faithfully preserves information in the data.

We first introduce a set of layer-wise decoders for both videos and texts. The key idea is to pair the encoders with decoders so that each pair of functions is an auto-encoder. Specifically, the decoder is also a layer of LSTM/GRU units, generating sequences of data. Thus, at the level of video (or paragraph), we will have a decoder to generate clips (or sentences). And at the level of clips (or sentences), we will have a decoder to generate frames (or words). Concretely, we would like to minimize the difference between what are generated by the decoders and what are computed by encoders on the data. Let

$$\{\hat{\mathbf{c}}_i\} = \text{DEC}_v^{(2)}(\mathbf{v}), \{\hat{\mathbf{s}}_i\} = \text{DEC}_p^{(2)}(\mathbf{p}) \quad (8)$$

be the two (high-level) decoders for videos and texts respectively. And similarly, for the decoder at the low-level

$$\{\hat{\mathbf{x}}_{ij}\} = \text{DEC}_v^{(1)}(\hat{\mathbf{c}}_i), \{\hat{\mathbf{w}}_{ij}\} = \text{DEC}_p^{(1)}(\hat{\mathbf{s}}_i) \quad (9)$$

where the low-level decoders take each *generated* clip and sentence embeddings as inputs and output sequences of generated frame and word embeddings.

$$\begin{aligned} \ell_{\text{RECONSTRUCT}}(v, p) = & \sum_i \{ \|\hat{\mathbf{c}}_i - \mathbf{c}_i\|_2^2 + \frac{1}{n_i} \sum_j \|\hat{\mathbf{x}}_{ij} - \mathbf{x}_{ij}\|_2^2 \} \\ & + \sum_i \{ \|\hat{\mathbf{s}}_i - \mathbf{s}_i\|_2^2 + \frac{1}{n'_i} \sum_j \|\hat{\mathbf{w}}_{ij} - \mathbf{w}_{ij}\|_2^2 \} \end{aligned} \quad (10)$$

Using those generated embeddings, we can construct a loss function characterizing how well the encoders encode the data pair  $(v, p)$  (see Eq 10).

### 3.4 Final Learning Objective and Its Extensions

The final learning objective is to balance all those loss quantities

$$\ell = \ell^{\text{HIGH}} + \ell^{\text{LOW}} + \tau \sum_k \ell_{\text{RECONSTRUCT}}(\mathbf{v}_k, \mathbf{p}_k) \quad (11)$$

where the high-level and low-level losses are defined as

$$\ell^{\text{HIGH}} = \ell_{\text{MATCH}}^{\text{HIGH}} + \ell_{\text{CLUSTER}}^{\text{HIGH}}, \quad \ell^{\text{LOW}} = \ell_{\text{MATCH}}^{\text{LOW}} + \ell_{\text{CLUSTER}}^{\text{LOW}} \quad (12)$$

In our experiments, we will study the contribution by each term.

**Learning under Weak Correspondences.** Our idea can be also extended to the common setting where only high-level alignments are available. In fact, high-level coarse alignments of data are easier and more economical to obtain, compared to fine-grained alignments between each sub-level sentence and video clip.

Since we do not have enough information to define the low-level matching loss  $\ell_{\text{MATCH}}^{\text{LOW}}$  exactly, we resort to approximation. We first define an averaged matching over all pairs of clips and sentences for a pair of video and paragraph

$$\overline{\text{MATCH}}(v, p) = \frac{1}{nm} \sum_{c_i} \sum_{s_j} \text{MATCH}(c_i, s_j) \quad (13)$$

where we relax the assumption that there is precisely the same number of sentences and clips. We use this averaged quantity to approximate the low-level matching loss

$$\begin{aligned} \tilde{\ell}_{\text{MATCH}}^{\text{LOW}} = & \sum_k \sum_{k' \neq k} [\beta' + \overline{\text{MATCH}}(\mathbf{v}_k, \mathbf{p}_k) - \overline{\text{MATCH}}(\mathbf{v}_{k'}, \mathbf{p}_k)]_+ \\ & + [\beta' + \overline{\text{MATCH}}(\mathbf{v}_k, \mathbf{p}_k) - \overline{\text{MATCH}}(\mathbf{v}_k, \mathbf{p}_{k'})]_+ \end{aligned} \quad (14)$$

This objective will push a clip embedding closer to the embeddings of the sentences belonging to the corresponding video (and vice versa for sentences to the corresponding video). A more refined approximation involving a soft assignment of matching can also be derived, which will be left for future work.

## 4 Experiments

We evaluate and demonstrate the advantage of learning hierarchical cross-modal embedding with our proposed approach on several tasks: (i) large-scale video-paragraph retrieval (Section 4.2), (ii) down-stream tasks such as video captioning (Section 4.3), and (iii) action recognition (Section 4.4).

### 4.1 Experiment Setups

**Datasets.** We evaluate on three large-scale video datasets:

(1) **ActivityNet Dense Caption** [17]. This variant of ActivityNet contains densely labeled temporal segments for 10,009 training and 4,917/4,885 (val1/val2) validation videos. Each video contains multiple clips and a corresponding paragraph with sentences aligned to the clips. In all our retrieval experiments, we follow the setting in [17] and report retrieval metrics such as recall@k (k=1,5,50) and median rank (MR). Following [17] we use ground-truth clip proposals as input for our main results. In addition, we also study our algorithm with a heuristic proposal method (see Section 4.2). In the main text, we report all results on validation set 1 (val1). Please refer to the Supp. Material for the results on val2. For video caption experiment, we follow [17] and evaluate on the validation set (val1 and val2). Instead of using action proposal method, ground-truth video segmentation is used for training and evaluation. Performances are reported in Bleu@K, METEOR and CIDEr.

(2) **DiDeMo** [1]. The original goal of DiDeMo dataset is to locate the temporal segments that correspond to unambiguous natural language descriptions in a video. We re-purpose it for the task of video and paragraph retrieval. It contains 10,464 videos, 26,892 video clips and 40,543 sentences. The training, validation and testing split contain 8,395, 1,065 and 1,004 videos and corresponding paragraphs, respectively. Each video clip may correspond to one or many sentences. For the video and paragraph retrieval task, paragraphs are constructed by concatenating all sentences that corresponding to one video. Similar to the setting in ActivityNet, we use the ground-truth clip proposals as input.

(3) **ActivityNet Action Recognition** [9]. We use ActivityNet V1.3 for aforementioned off-the-shelf action recognition. The dataset contains 14,950 untrimmed videos with 200 action classes, which is split into training and validation set. Training and validation set have 10,024 and 4,926 videos, respectively. Among all 200 action classes, 189 of the action classes have been covered by the vocabulary extracted from the paragraph corpus and 11 of the classes are unseen.

**Baselines and Our Methods.** We use the FSE method (as described in Section 3.1) as a baseline model. It ignores the clip and sentence structures in the videos and paragraphs. We train a one-layer GRU directly on the extracted frame/word features and take their outputs as the embedding representing each modality. Results with C3D features are also included (see Table 1).

Our method has two variants: when  $\tau = 0$ , the method ( $\text{HSE}_{[\tau=0]}$ ) simplifies to a stacked/hierarchical sequence models as used in [19, 25] except that they



**Table 1.** Video paragraph retrieval on ActivityNet (val1). Standard deviation from 3 random seeded experiments are also reported.

	Paragraph $\Rightarrow$ Video				Video $\Rightarrow$ Paragraph			
	R@1	R@5	R@50	MR	R@1	R@5	R@50	MR
C3D Feature with Dimensionality Reduction [33]								
LSTM-YT [35]	0.0	4.0	24.0	102.0	0.0	7.0	38.0	98.0
NO CONTEXT [36]	5.0	14.0	32.0	78.0	7.0	18.0	45.0	56.0
DENSE online[17]	10.0	32.0	60.0	36.0	17.0	34.0	70.0	33.0
DENSE full[17]	14.0	32.0	65.0	34.0	18.0	36.0	74.0	32.0
FSE	12.6 $\pm$ 0.4	33.2 $\pm$ 0.3	77.6 $\pm$ 0.3	12.0	11.5 $\pm$ 0.5	31.8 $\pm$ 0.3	77.7 $\pm$ 0.3	13.0
HSE[ $\tau=0$ ]	32.8 $\pm$ 0.3	62.3 $\pm$ 0.4	90.5 $\pm$ 0.1	3.0	32.0 $\pm$ 0.6	62.5 $\pm$ 0.5	90.5 $\pm$ 0.3	3.0
HSE[ $\tau=5e-4$ ]	32.7 $\pm$ 0.7	63.2 $\pm$ 0.4	90.8 $\pm$ 0.2	3.0	32.8 $\pm$ 0.4	63.2 $\pm$ 0.2	91.2 $\pm$ 0.3	3.0
Inception-V3 pre-trained on Kinetics [40]								
FSE	18.2 $\pm$ 0.2	44.8 $\pm$ 0.4	89.1 $\pm$ 0.3	7.0	16.7 $\pm$ 0.8	43.1 $\pm$ 1.1	88.4 $\pm$ 0.3	7.3
HSE[ $\tau=0$ ]	43.9 $\pm$ 0.6	75.8 $\pm$ 0.2	96.9 $\pm$ 0.3	2.0	43.3 $\pm$ 0.6	75.3 $\pm$ 0.6	96.6 $\pm$ 0.2	2.0
HSE[ $\tau=5e-4$ ]	<b>44.4<math>\pm</math>0.5</b>	<b>76.7<math>\pm</math>0.3</b>	<b>97.1<math>\pm</math>0.1</b>	<b>2.0</b>	<b>44.2<math>\pm</math>0.6</b>	<b>76.7<math>\pm</math>0.3</b>	<b>97.0<math>\pm</math>0.3</b>	<b>2.0</b>

do not consider cross-modal learning with cross-modal matching loss while we do. We consider this as a very strong baseline. When  $\tau \neq 0$ , the HSE takes full advantage of layer-wise reconstruction with multiple decoders, at different levels of the hierarchy. In our experiments, this method gives the best results.

**Implementation Details.** Following the settings of [17], we extract the C3D features [33] pretrained on Sports-1M dataset [13] for raw videos in ActivityNet. PCA is then used to reduce the dimensionality of the feature to 500. To verify the generalization of our model across different sets of visual feature, as well as leveraging the state-of-the-art video models, we also employed recently proposed TSN-Inception V3 network [38] pre-trained on Kinetics [14] dataset to extract visual features. Similarly, we extract TSN-Inception V3 feature for videos in Didemo dataset. We do not fine-tuning the convolutional neural network on the video along the training to reduce the computational cost. For word embedding, we use 300 dimension GloVe [27] features pre-trained on 840B common web-crawls. In all our experiments, we use GRU as sequence encoders. For HSE, we choose  $\tau = 0.0005$  from tuning this hyper-parameter on the val2 set of ActivityNet retrieval dataset. The same  $\tau$  value is used for experiments on DiDeMo, without further tuning. (More details in the Supp. Material)

## 4.2 Results on Video-Paragraph Retrieval

In this section, we first compare our proposed approach to the state-of-the-art algorithms, and then perform ablation studies on variants of our method, to evaluate the proposed learning objectives.

**Main Results.** We reported our results on ActivityNet Dense Caption val1 set and DiDeMo test set as Table 1 and Table 2, respectively. For both C3D

**Table 2.** Video paragraph retrieval on DiDeMo dataset. s2vT method is re-implemented for retrieval task.

	Paragraph $\Rightarrow$ Video				Video $\Rightarrow$ Paragraph			
	R@1	R@5	R@50	MR	R@1	R@5	R@50	MR
s2vT [36]	11.9	33.6	76.5	13.0	13.2	33.6	76.5	15.0
FSE	13.9 $\pm$ 0.7	36.0 $\pm$ 0.8	78.9 $\pm$ 1.6	11.0	13.1 $\pm$ 0.5	33.9 $\pm$ 0.4	78.0 $\pm$ 0.8	12.0
HSE[ $\tau=0$ ]	<b>30.2<math>\pm</math>0.8</b>	<b>60.5<math>\pm</math>1.1</b>	91.8 $\pm$ 0.7	3.3	29.4 $\pm$ 0.4	58.9 $\pm$ 0.7	91.9 $\pm$ 0.6	3.7
HSE[ $\tau=5e-4$ ]	29.7 $\pm$ 0.2	60.3 $\pm$ 0.9	<b>92.4<math>\pm</math>0.3</b>	<b>3.3</b>	<b>30.1<math>\pm</math>1.2</b>	<b>59.2<math>\pm</math>0.9</b>	<b>92.1<math>\pm</math>0.5</b>	<b>3.0</b>

**Table 3.** Ablation studies on the learning objectives.

Dataset	$\ell^{\text{LOW}}$	Paragraph $\Rightarrow$ Video			Video $\Rightarrow$ Paragraph			
		R@1	R@5	R@50	R@1	R@5	R@50	
ActivityNet	$\times$	41.8 $\pm$ 0.4	74.1 $\pm$ 0.6	96.6 $\pm$ 0.1	40.5 $\pm$ 0.4	73.9 $\pm$ 0.6	96.3 $\pm$ 0.1	
	HSE[ $\tau=0$ ]	WEAK	42.6 $\pm$ 0.4	74.8 $\pm$ 0.3	96.7 $\pm$ 0.1	41.3 $\pm$ 0.2	74.7 $\pm$ 0.4	96.5 $\pm$ 0.1
		STRONG	43.9 $\pm$ 0.6	75.8 $\pm$ 0.2	96.9 $\pm$ 0.3	43.3 $\pm$ 0.6	75.3 $\pm$ 0.6	96.6 $\pm$ 0.2
	HSE[ $\tau=5e-4$ ]	$\times$	42.5 $\pm$ 0.3	74.8 $\pm$ 0.1	96.9 $\pm$ 0.0	41.6 $\pm$ 0.2	74.7 $\pm$ 0.6	96.6 $\pm$ 0.1
		WEAK	43.0 $\pm$ 0.6	75.2 $\pm$ 0.4	96.9 $\pm$ 0.1	41.5 $\pm$ 0.1	75.2 $\pm$ 0.6	96.8 $\pm$ 0.2
		STRONG	44.4 $\pm$ 0.5	76.7 $\pm$ 0.3	97.1 $\pm$ 0.1	44.2 $\pm$ 0.6	76.7 $\pm$ 0.3	97.0 $\pm$ 0.3
DiDeMo	$\times$	27.1 $\pm$ 1.9	59.1 $\pm$ 0.4	92.2 $\pm$ 0.3	27.3 $\pm$ 1.0	57.6 $\pm$ 0.5	91.3 $\pm$ 1.2	
	HSE[ $\tau=0$ ]	WEAK	28.0 $\pm$ 0.8	58.9 $\pm$ 0.5	91.4 $\pm$ 0.6	28.3 $\pm$ 0.3	58.5 $\pm$ 0.6	91.2 $\pm$ 0.3
		STRONG	30.2 $\pm$ 0.8	60.5 $\pm$ 1.1	91.8 $\pm$ 0.7	29.4 $\pm$ 0.4	58.9 $\pm$ 0.7	91.9 $\pm$ 0.6
	HSE[ $\tau=5e-4$ ]	$\times$	28.1 $\pm$ 0.8	59.5 $\pm$ 1.1	91.7 $\pm$ 0.7	28.2 $\pm$ 0.8	58.1 $\pm$ 0.5	90.9 $\pm$ 0.5
		WEAK	28.7 $\pm$ 2.1	59.1 $\pm$ 0.2	91.6 $\pm$ 0.7	28.3 $\pm$ 0.8	59.2 $\pm$ 0.6	91.1 $\pm$ 0.1
		STRONG	29.7 $\pm$ 0.2	60.3 $\pm$ 0.9	92.4 $\pm$ 0.3	30.1 $\pm$ 1.2	59.2 $\pm$ 0.9	92.1 $\pm$ 0.5

and Inception V3 feature, we observed performances on our hierarchical models improved the previous state-of-the-art result by a large margin (on Recall@1, over  $\sim 15\%$  improvement with C3D and  $\sim 30\%$  improvement with InceptionV3). DENSE full [17], which models the flat sequences of clips, outperforms our FSE baseline as they augment each segment embedding with a weighted aggregated context embedding. However, it fails to model more complex temporal structures of video and paragraph, which leads to inferior performance to our HSE models.

Comparing to our flat baseline model, both HSE[ $\tau=0$ ] and HSE[ $\tau=5e-4$ ] improve performances over all metrics in retrieval. It implies that hierarchical modeling can effectively capture the structure information and relationships over clips and sentences among videos and paragraphs. Moreover, we observe that HSE[ $\tau=5e-4$ ] consistently improves over HSE[ $\tau=0$ ] across most retrieval metrics on both datasets. This attributes the importance of our layer-wise reconstruction objectives, which suggests that better generalization performances.

**Low-level Loss is Beneficial.** Table 1 and Table 2 have shown results with optimizing both low-level and high-level objectives. In Table 3, we further performed ablation studies on the learning objectives. Note that rows with  $\times$  represent learning without low-level loss  $\ell^{\text{LOW}}$ . In all scenarios, joint learning with both low-level and high-level correspondences improves the retrieval performance.

**Table 4.** Performance of using proposal instead of ground truth on ActivityNet dataset

Proposal Method	# Segments	$\mathbf{P} \Rightarrow \mathbf{V}$		$\mathbf{V} \Rightarrow \mathbf{P}$		Precision	Recall
		R@1	R@5	R@1	R@5		
HSE + SSN	-	10.4	31.9	10.8	31.7	1.5	17.1
HSE + UNIFORM	1	18.0	45.5	16.5	44.9	63.2	31.1
	2	20.0	48.9	18.4	47.6	61.8	46.0
	3	20.0	48.6	18.2	47.9	55.3	50.6
	4	20.5	49.3	18.7	48.1	43.2	45.5
HSE + GROUND TRUTH	-	44.4	76.7	44.2	76.7	100.0	100.0
FSE	-	18.2	44.8	16.7	43.1	-	-

**Learning with Weak Correspondences at Low-level.** As mentioned in Section 3, our method can be extended to learn the low-level embedding with weak correspondence. We evaluate its effectiveness on both ActivityNet and DiDeMo datasets. Performance are listed in Table 3. Note that for the rows of “weak”, no auxiliary alignments between sentences and clips are available during training.

Clearly, including low-level loss with weak correspondence (ie, correspondence only at the high-level) obtained superior performances when compared to models that do not include low-level loss at all. On several occasions, it even attains the same competitive result as including low-level loss with strong correspondences at the clip/sentence levels.

**Learning with Video Proposal Methods.** As using ground-truth temporal segments of videos is not a natural assumption, we perform experiments to validate the effectiveness of our method with proposal methods. Specifically, we experiment with two different proposal approaches: SSN [46] pre-trained on ActivityNet action proposal and a heuristic uniform proposal. For uniform proposal of K segments, we meant naturally segmenting a video into K non-overlapping and equal-length temporal segments.

The results are summarized in Table 4 (with columns of precision and recall being the performance metrics of the proposal methods). There are two main conclusions from these results: (1) The segments of Dense Caption dataset deviate significantly from the action proposals, therefore a pre-trained action proposal algorithm performs poorly. (2) Even with heuristic proposal methods, the performance of HSE is mostly better than (or comparable with) FSE. We leave to future work on identifying stronger methods for proposals.

**Retrieval with Incomplete Video and Paragraph.** In this section, we investigate the correlation between the number of observed clips and sentences and models’ performance of video and paragraph retrieval. In this experiment, we gradually increase the number of clips and sentences observed by our model during the testing and obtained the Figure 4, on ActivityNet. When the video/paragraph contains fewer clips/sentences than the number of observations we required, we

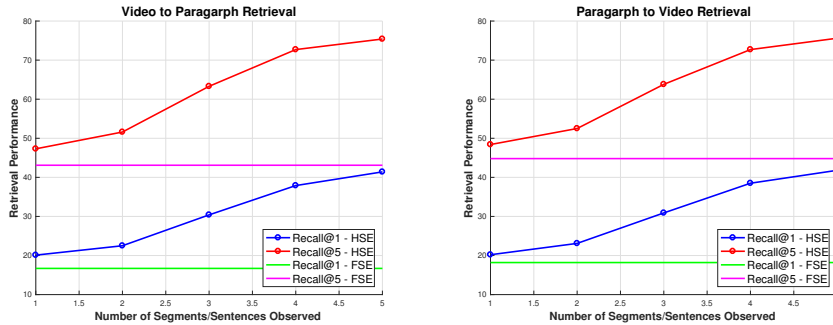


Fig. 4. Retrieval performance improves given more observed clips/sentences.

take all those available clips/sentences for computing the video/paragraph embedding. (On average 3.65 clips/sentences per video/paragraph)

From Figure 4, we note that increasing the number of the observed clips and sentences leads to improved performance results in retrievals. We can see that when observing only one clip and sentence, our model already outperforms the previous state-of-the-art method as well as our baseline FSE that observes the entire sequence. With observing less than the average length of clips and sentences, our learned model can achieve  $\sim 70\%$  of the final performance.

### 4.3 Results on Video Captioning

**Setups.** In addition to the video paragraph retrieval, we evaluate our learned embeddings for video captioning. Specifically, we follow [17] and train a caption model [37] on top of the pre-trained video embeddings. Similar to [17], we concatenate the clip-level feature with contextual video-level feature, and build a two-layer LSTM as a caption generator. We randomly initialized the word embedding as well as LSTM and trained the model for 25 epochs with learning rate of 0.001. We use the ground-truth proposal throughout training and evaluation following the setting of [17, 20]. During testing, beam search is used with beam=5. Results are reported in Table 5.

**Results.** We observe that our proposed model outperforms baseline over most metrics. Meanwhile, HSE also improves over previous approaches such as LSTM-YT, s2VT, and HRNN on B@2, METEOR, and CIDEr by a margin. HSE achieves comparable results with DVC in all criterions. However, both HSE and HSE $[\tau=0]$  failed to obtain close performance to DENSE [17]. This may due to the fact that DENSE [17] carefully learns to aggregate the context information of a video clip for producing high-quality caption, while optimized for video-paragraph retrieval our embedding model does not equip with such capability. However, it is worth noting that our model obtains higher CIDEr score compared to all

**Table 5.** Results for video captioning on ActivityNet

	B@1	B@2	B@3	B@4	M	C
LSTM-YT [35]	18.2	7.4	3.2	1.2	6.6	14.9
S2VT [36]	20.4	9.0	4.6	2.6	7.9	21.0
HRNN [43]	19.5	8.8	4.3	2.5	8.0	20.2
DENSE [17]	26.5	13.5	7.1	4.0	9.5	24.6
DVC [20]	19.6	9.9	4.6	1.6	10.3	25.2
FSE	17.9	8.2	3.6	1.7	8.7	32.1
HSE[ $\tau=0$ ]	19.6	9.4	4.2	2.0	9.2	39.5
HSE[ $\tau=5e-4$ ]	19.8	9.4	4.3	2.1	9.2	39.8

**Table 6.** Results for action recognition on ActivityNet (low-level embeddings)

	Zero-Shot Transfer		Train Classifier	
	Top-1	Top-5	Top-1	Top-5
FV-VAE [28]	-	-	78.6	-
TSN [39]	-	-	88.1	-
FSE	48.3	79.4	74.4	94.1
HSE[ $\tau=0$ ]	50.2	84.4	74.7	94.3
HSE[ $\tau=5e-4$ ]	51.4	83.8	75.3	94.3
RANDOM	0.5	2.5	0.5	2.5

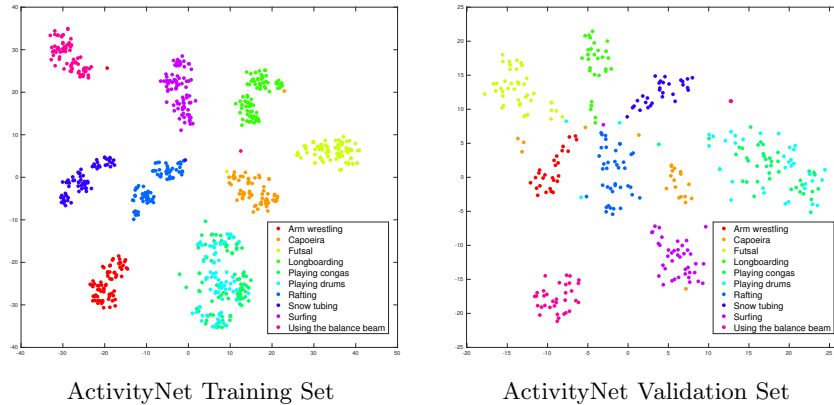
existing methods. We empirically observe that fine-tuning the pre-trained video embedding does not lead to further performance improvement.

#### 4.4 Results on Action Recognition

To evaluate the effectiveness of our model, we take the off-the-shelf clip-level embeddings trained on video-paragraph retrieval for action recognition (on ActivityNet with non-overlapping training and validation data). We use two action recognition settings to evaluate, namely **zero-shot transfer** and **classification**.

**Setups.** In the **zero-shot** setting, we directly evaluate our low-level embedding model learned in the video and text retrieval, via treating the phrases of actions as sentences and use the sentence-level encoder to encode the action embedding. We take the raw video and apply clip-level video encoder to extract the feature for retrieving actions. No re-training is performed and all models have no access to the actions’ data distribution. Note though action are not directly used as sentences during the training, some are available as verbs in the vocabulary. Meanwhile, as we are using pre-trained word vector (GloVe), it allows the transfer to unseen actions. In the **classification** setting, we discriminatively train a simple classifier to measure the classification accuracy. Concretely, a one-hidden-layer Multi-Layer Perceptron (MLP) is trained on the clip-level embeddings. We do not fine-tune the pre-trained clip-level video embedding here.

**Results.** We report results of above two settings on the ActivityNet validation set (see Table 6). We observe that our learned low-level embeddings allow superior zero-shot transfer to action recognition, without accessing any training data. This indicates that semantics of actions are indeed well reserved in the learned embedding models. More interestingly, we can see that both HSE[ $\tau=0$ ] and HSE improve the performance over FSE. It shows that our hierarchical modeling of video benefits not only high-level embedding but also low-level embedding. A similar trend is also observed in the classification setting. Our method achieves comparable performance to the state-of-the-art video modeling approach such as FV-VAE [28]. Note TSN [39] is fully supervised thus not directly comparable.



**Fig. 5.** T-SNE visualization of off-the-shelf video embedding of HSE on ActivityNet v1.3 training and validation set. Points are marked with its action classes.

#### 4.5 Qualitative Results

We use t-SNE [23] to visualize our results in the video to paragraph and paragraph to video retrieval task. Fig 5 shows that the proposed method can cluster the embedding of videos with regard to its action classes. To further explain the retrieval quality, we provide qualitative visualization in the Supp. Material.

## 5 Conclusion

In this paper, we propose a novel cross-modal learning approach to model videos and texts jointly, which leverages the intrinsic hierarchical structures of both videos or texts. Specifically, we consider the correspondences of videos and texts at multiple granularities, and derived loss functions to align the embeddings for the paired clips and sentences, as well as paired video and paragraph in accordance in their own semantic spaces. Another important component of our model is layer-wise reconstruction, which ensures that learned embeddings capture video (paragraph) and clips (words) at different levels. Moreover, we further extend our learning objective so that it allows to handle a more generalized learning scenario where only video paragraph correspondence exists. We demonstrate the advantage of our proposed model in a range of tasks including video and text retrieval, zero-shot action recognition and video caption.

**Acknowledgments** We appreciate the feedback from the reviewers. This work is partially supported by NSF IIS-1065243, 1451412, 1513966/ 1632803/1833137, 1208500, CCF-1139148, a Google Research Award, an Alfred P. Sloan Research Fellowship, gifts from Facebook and Netflix, and ARO# W911NF-12-1-0241 and W911NF-15-1-0484.

## References

1. Anne Hendricks, L., Wang, O., Shechtman, E., Sivic, J., Darrell, T., Russell, B.: Localizing moments in video with natural language. In: ICCV. pp. 5804–5813
2. Antol, S., Agrawal, A., Lu, J., Mitchell, M., Batra, D., Lawrence Zitnick, C., Parikh, D.: Vqa: Visual question answering. In: ICCV. pp. 2425–2433 (2015)
3. Chao, W.L., Hu, H., Sha, F.: Being negative but constructively: Lessons learnt from creating better visual question answering datasets. NAACL-HLT pp. 431–441 (2018)
4. Chung, J., Gulcehre, C., Cho, K., Bengio, Y.: Empirical evaluation of gated recurrent neural networks on sequence modeling. arXiv preprint arXiv:1412.3555 (2014)
5. Collell, G., Moens, M.F.: Is an image worth more than a thousand words? on the fine-grain semantic differences between visual and linguistic representations. In: COLING. pp. 2807–2817 (2016)
6. Feichtenhofer, C., Pinz, A., Zisserman, A.: Convolutional two-stream network fusion for video action recognition. In: CVPR. pp. 1933–1941 (2016)
7. Frome, A., Corrado, G.S., Shlens, J., Bengio, S., Dean, J., Mikolov, T., et al.: Devise: A deep visual-semantic embedding model. In: NIPS. pp. 2121–2129 (2013)
8. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR. pp. 770–778 (2016)
9. Heilbron, F.C., Escorcia, V., Ghanem, B., Niebles, J.C.: Activitynet: A large-scale video benchmark for human activity understanding. In: CVPR. pp. 961–970
10. Hochreiter, S., Schmidhuber, J.: Long short-term memory. *Neural computation* **9**(8), 1735–1780 (1997)
11. Hu, H., Chao, W.L., Sha, F.: Learning answer embeddings for visual question answering. In: CVPR. pp. 5428–5436 (2018)
12. Karpathy, A., Fei-Fei, L.: Deep visual-semantic alignments for generating image descriptions. In: CVPR. pp. 3128–3137 (2015)
13. Karpathy, A., Toderici, G., Shetty, S., Leung, T., Sukthankar, R., Fei-Fei, L.: Large-scale video classification with convolutional neural networks. In: CVPR. pp. 1725–1732 (2014)
14. Kay, W., Carreira, J., Simonyan, K., Zhang, B., Hillier, C., Vijayanarasimhan, S., Viola, F., Green, T., Back, T., Natsev, P., et al.: The kinetics human action video dataset. arXiv preprint arXiv:1705.06950 (2017)
15. Kiela, D., Bottou, L.: Learning Image Embeddings using Convolutional Neural Networks for Improved Multi-Modal Semantics. In: EMNLP. pp. 36–45 (2014)
16. Kiros, R., Salakhutdinov, R., Zemel, R.S.: Unifying visual-semantic embeddings with multimodal neural language models. arXiv preprint arXiv:1411.2539 (2014)
17. Krishna, R., Hata, K., Ren, F., Fei-Fei, L., Niebles, J.C.: Dense-captioning events in videos. In: ICCV. pp. 706–715 (2017)
18. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: NIPS. pp. 1106–1114 (2012)
19. Li, J., Luong, M.T., Jurafsky, D.: A hierarchical neural autoencoder for paragraphs and documents. ACL pp. 1106–1115 (2015)
20. Li, Y., Yao, T., Pan, Y., Chao, H., Mei, T.: Jointly localizing and describing events for dense video captioning. In: CVPR. pp. 7492–7500 (2018)
21. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: ECCV. pp. 740–755 (2014)

22. Luong, M.T., Pham, H., Manning, C.D.: Effective approaches to attention-based neural machine translation. *EMNLP* pp. 1412–1421
23. Maaten, L.v.d., Hinton, G.: Visualizing data using t-sne. *JMLR* **9**(Nov), 2579–2605 (2008)
24. Niu, Z., Zhou, M., Wang, L., Gao, X., Hua, G.: Hierarchical multimodal lstm for dense visual-semantic embedding. In: *ICCV*. pp. 1899–1907 (2017)
25. Pan, P., Xu, Z., Yang, Y., Wu, F., Zhuang, Y.: Hierarchical recurrent neural encoder for video representation with application to captioning. In: *CVPR*. pp. 1029–1038 (2016)
26. Pascanu, R., Mikolov, T., Bengio, Y.: On the difficulty of training recurrent neural networks. In: *ICML*. pp. 1310–1318 (2013)
27. Pennington, J., Socher, R., Manning, C.D.: Glove: Global vectors for word representation. In: *EMNLP*. pp. 1532–1543 (2014)
28. Qiu, Z., Yao, T., Mei, T.: Deep quantization: Encoding convolutional activations with deep generative model. In: *CVPR*. pp. 4085–4094 (2017)
29. Schroff, F., Kalenichenko, D., Philbin, J.: Facenet: A unified embedding for face recognition and clustering. In: *CVPR*. pp. 815–823 (2015)
30. Simonyan, K., Zisserman, A.: Two-stream convolutional networks for action recognition in videos. In: *NIPS*. pp. 568–576 (2014)
31. Soomro, K., Zamir, A.R., Shah, M.: Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402* (2012)
32. Sutskever, I., Vinyals, O., Le, Q.V.: Sequence to sequence learning with neural networks. In: *NIPS*. pp. 3104–3112 (2014)
33. Tran, D., Bourdev, L., Fergus, R., Torresani, L., Paluri, M.: Learning spatiotemporal features with 3d convolutional networks. In: *ICCV*. pp. 4489–4497 (2015)
34. Tsai, Y.H.H., Huang, L.K., Salakhutdinov, R.: Learning robust visual-semantic embeddings. In: *ICCV*. pp. 3591–3600 (2017)
35. Venugopalan, S., Rohrbach, M., Donahue, J., Mooney, R., Darrell, T., Saenko, K.: Sequence to sequence-video to text. In: *ICCV*. pp. 4534–4542 (2015)
36. Venugopalan, S., Xu, H., Donahue, J., Rohrbach, M., Mooney, R., Saenko, K.: Translating videos to natural language using deep recurrent neural networks. *NAACL-HLT* pp. 1494–1504 (2015)
37. Vinyals, O., Toshev, A., Bengio, S., Erhan, D.: Show and tell: A neural image caption generator. In: *CVPR*. pp. 3156–3164 (2015)
38. Wang, L., Xiong, Y., Wang, Z., Qiao, Y., Lin, D., Tang, X., Van Gool, L.: Temporal segment networks: Towards good practices for deep action recognition. In: *ECCV*. pp. 20–36 (2016)
39. Wang, L., Xiong, Y., Wang, Z., Qiao, Y., Lin, D., Tang, X., Van Gool, L.: Temporal segment networks for action recognition in videos. *arXiv preprint arXiv:1705.02953* (2017)
40. Wang, L., Li, Y., Lazebnik, S.: Learning deep structure-preserving image-text embeddings. In: *CVPR*. pp. 5005–5013 (2016)
41. Wu, C.Y., Zaheer, M., Hu, H., Manmatha, R., Smola, A.J., Krähenbühl, P.: Compressed video action recognition. *CVPR* (2018)
42. Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A., Salakhutdinov, R., Zemel, R., Bengio, Y.: Show, attend and tell: Neural image caption generation with visual attention. In: *ICML*. pp. 2048–2057 (2015)
43. Yu, H., Wang, J., Huang, Z., Yang, Y., Xu, W.: Video paragraph captioning using hierarchical recurrent neural networks. In: *CVPR*. pp. 4584–4593 (2016)
44. Zhang, B., Wang, L., Wang, Z., Qiao, Y., Wang, H.: Real-time action recognition with enhanced motion vector cnns. In: *CVPR*. pp. 2718–2726 (2016)



45. Zhang, K., Chao, W.L., Sha, F., Grauman, K.: Video summarization with long short-term memory. In: ECCV. pp. 766–782 (2016)
46. Zhao, Y., Xiong, Y., Wang, L., Wu, Z., Tang, X., Lin, D.: Temporal action detection with structured segment networks. ICCV pp. 2933–2942 (2017)
47. Zhu, Y., Groth, O., Bernstein, M., Fei-Fei, L.: Visual7w: Grounded question answering in images. In: CVPR. pp. 4995–5004 (2016)