LoGAN: Latent Graph Co-Attention Network for Weakly-Supervised Video Moment Retrieval

Reuben Tan
Boston University
rxtan@bu.edu

Huijuan Xu
University of California, Berkeley
hxu@berkeley.edu

Kate Saenko
Boston University
MIT-IBM Watson AI Lab
saenko@bu.edu

Bryan A. Plummer
Boston University
bplum@bu.edu

Abstract

The goal of weakly-supervised video moment retrieval is to localize the video segment most relevant to a description without access to temporal annotations during training. Prior work uses co-attention mechanisms to understand relationships between the vision and language data, but they lack contextual information between video frames that can be useful to determine how well a segment relates to the query. To address this, we propose an efficient Latent Graph Co-Attention Network (LoGAN) that exploits fine-grained frame-by-word interactions to jointly reason about the correspondences between all possible pairs of frames, providing context cues absent in prior work. Experiments on the DiDeMo and Charades-STA datasets demonstrate the effectiveness of our approach, where we improve Recall@1 by 5-20% over prior weakly-supervised methods, even boasting an 11% gain over strongly-supervised methods on DiDeMo, while also using significantly fewer model parameters than other co-attention mechanisms.

1. Introduction

The task of *video moment retrieval* is to temporally localize a "moment" or event in a video given the linguistic description of that event. To avoid costly annotation of start and end frames for each event, *weakly-supervised* moment retrieval methods learn a mapping of latent correspondences between the visual and linguistic elements [25, 20, 11]. As illustrated in Fig. 1(a), these methods use features representing the entire input sentence in order to learn the importance of each frame to the query image. While some methods go a step further and use a co-attention mechanism that reasons about the vision and language features together to identify important features in both modalities (*e.g.* [3, 19, 24, 40]), they have been developed for other tasks and settings that may not generalize to weakly-supervised moment retrieval.

In this paper, we propose a Latent Graph Co-Attention Network (LoGAN), a novel co-attention model that identifies the video segment related to a descriptive sentence by performing fine-grained semantic reasoning over an entire video. We begin by learning which words in the query are important to frames in the video in order to obtain a visual-semantic representation for each word. Then, we use this representation to learn how important each frame is to the query using a Word-Conditioned Visual Graph (WCVG). This graph iteratively updates the frame features using messages passed from the visual-semantic features, then we update the visual-semantic features using the new frame features for the next iteration. Thus, as illustrated in Fig. 1(b), each frame has been updated with information from the query as well as all the other frames via the visual-semantic features.

To ensure important temporal information is not lost, prior work used the starting and stopping location of each segment that they referred to as temporal endpoint features [14]. Instead, we encode the relative position of each frame's features by concatenating them with positional encodings [34], which we found work far better. To the best of our knowledge, this is the first application of these encodings to the moment retrieval task. Our approach is trained end-to-end using a triplet loss, where we encourage a video and its ground truth description to embed closer than a description randomly sampled from another video (*i.e.*, no video segments are used during training). Following [25], at test time, we use LoGAN and the text query to rank video segments generated via a sliding window approach.

As mentioned earlier, to date, none of the weakly-supervised moment retrieval methods [25, 20, 11] use co-attention, but rather just update their visual features using the text query (but not the other way around). Our approach is closest in spirit to the strongly-supervised MAN [40], which also uses a co-attention module whose visual representations are updated using a graph. However, they use correspondences between words and video segments rather than frames, which is troublesome since we do not have temporal annotations, which makes learning good segment

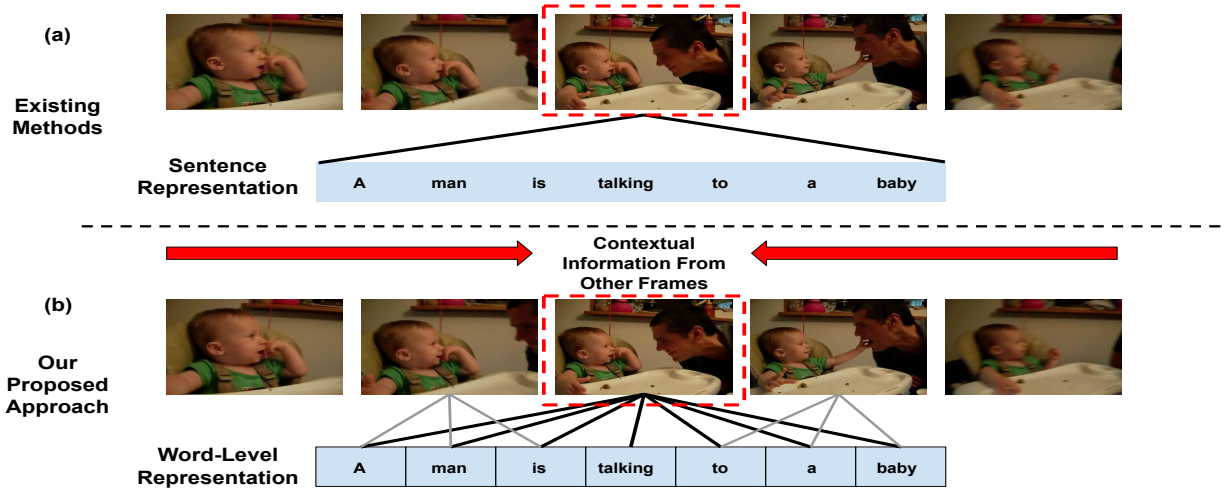

Figure 1. Given a video and a sentence, our aim is to retrieve the most relevant segment (enclosed by the red bounding box). Existing methods as shown in (a) consider video frames as independent inputs and ignore other frames in the video. In contrast, our proposed approach shown in (b) aggregates video contextual information from all the frames using graph propagation and leverages fine-grained frame-by-word interactions for more accurate semantic retrieval. (Only some interactions are shown to prevent overcrowding the figure.)

proposals difficult. Thus, as we will show, this leads to poor performance in the weakly-supervised setting. We also share some similarities to TGN [3], a strongly-supervised moment retrieval method that learns correspondences between frames and sentence features using a recurrent network. However, our weakly-supervised frame-by-word module that operates directly on the features in both modalities rather than using a recurrent network gets 5% better Recall@1 on the DiDeMo dataset [14], without even using our positional encodings or WCVG module. With our full model our Recall@1 performance is boosted by another 6% (11% better than TGN in total).

We summarize our contributions below:

- We propose a nov el Latent Co-Attention Network (LoGAN), which significantly improves the latent alignment between videos and natural language by leveraging a multi-level co-attention mechanism to learn contextualized visual-semantic representations, improving Recall@1 by 5-20% over prior work on the Charades-STA and DiDeMo datasets.
- We introduce a novel application of positional encodings to learn temporally-aware multimodal features.
- We provide a thorough analysis of video-language interaction modules to validate our contributions and provide a useful reference for future work.

2. Related Work

The most related work to ours are weakly-supervised moment retrieval methods [25, 20, 11]. [25] use a text-guided attention to identify important frames to the query. [20] use a sentence representation and frame features as input into a Transformer [34] to identify relevant segments. [11] decouple the task into alignment and detection components which are jointly learned in an end-to-end trained network. As we mentioned in the Introduction, these methods all use text to attend to important frames, but do not use the frames to identify important text features. In addition, WCVG enables us to further refine our co-attention predictions, resulting in significant performance gains over these methods.

Most of the recent work in video moment retrieval using natural language queries use strong supervision (*e.g.* [10, 14, 13, 38, 40, 3, 39, 6, 4, 12]), where the provided temporal annotations can be used to improve the alignment between the visual and language modalities. As mentioned in the Introduction, these methods rely on having temporal annotations during training, and often do not generalize to the weakly-supervised setting. In addition to the MAN and TGN methods we discussed earlier, another notable strongly-supervised work is DEBUG [22], which also uses co-attention over frames and words, but doesn't use positional encodings or perform joint reasoning over the entire video. Thus, our approach obtains similar performance as theirs without using temporal annotations during training.

Co-attention modules have also been used in other vision-language and audio [36] tasks. For example, [19] learn correspondences between words and image regions for image-text matching. [24] use n-grams and perform an alternating co-attention for visual question answering

(VQA) [1, 17, 2]. [15], who also address the VQA task, introduce a Language-Conditioned Graph Network (LCGN) that perform complex relational reasoning between words, sentences, and image regions. This is the most similar work to ours, but they learn relationships between words and image regions instead of words and frames as in LoGAN. However, these methods address tasks that operate on whole images or tens of image regions rather than the videos in our task that may have hundreds or thousands of frames and also requires temporal reasoning. Thus, as we will show, directly adapting their models to our task is less efficient and performs worse than our approach. Last but not least, they often form the core of general visual-and-language reasoning models [32, 23]. These reasoning models learn general representations that are applied to downstream tasks via transfer learning. Our MIL framework is also similar in nature to the Stacked Cross Attention Network (SCAN) model [19]. The SCAN model leverages image region-by-word interactions to learn better representations for image-text matching.

There are also a number of closely-related tasks to video moment retrieval such as temporal activity detection in videos. A general pipeline of proposal and classification is adopted by various temporal activity detection models [37, 41, 29] with the temporal proposals learnt by temporal coordinate regression. However, these approaches assume you are provided with a predefined list of activities, rather than an open-ended list provided via natural language queries at test time. Methods for visual phrase grounding also tend to be provided with natural language queries as input [5, 21, 9, 26, 16, 28], but the task is performed over image regions to locate a related bounding box rather than video segments to locate the correct moment.

3. Latent Graph Co-Attention Network

Given a video-sentence pair, the goal in video moment retrieval is to retrieve the most relevant video moment related to the description. In the weakly-supervised setting, we are provided with a video-sentence pair, but do not have access to the temporal annotations during training. To address this task, we introduce our Latent Graph Co-Attention Network (LoGAN), which learns contextualized visual-semantic representations from fine-grained frame-by-word interactions. As seen in Figure 2, our network has two major components - (1) we learn powerful representations conditioned on both modalities via Frame-By-Word attention which we concatenate with Positional Encodings [34] (details in Section 3.1), and (2) a Word-Conditioned Visual Graph where we update video frame representations based on context from the rest of the video, described in Section 3.2. These learned video frame representations are used to determine their relevance to their corresponding attended sentence representations using a LogSumExp (LSE) pooling similarity metric (described in Section 3.3).

3.1. Learning Tightly Coupled Multimodal Representations

In this section we discuss our initial video and sentence representations which are updated with contextual information in Section 3.2. We begin by encoding each word in an input sentence using a Gated Recurrent Unit (GRU) [7]. The output of this GRU is denoted as $W = \{w_1, w_2, \cdot \cdot \cdot, w_Q\}$ where Q is the number of words in the sentence. Then, each frame in the input video is encoded using a pre-trained Convolutional Neural Network (CNN). In the case of a 3D CNN this actually corresponds to a small chunk of sequential frames, but we shall refer to this as a frame representation throughout this paper for simplicity. The frame features are passed into a fully-connected layer followed by a ReLU layer. The outputs are concatenated with positional encodings (described below) to form the initial video representations, denoted as $V = \{v_1, v_2, ..., v_N\}$ where N is the number of frames in video V.

Positional Encodings (PE). To provide some notion of the relative position of each frame, we include the PE features which have been used in language tasks like learning language representations using BERT [8, 34]. These PE features can be thought of as similar to the temporal endpoint features (TEF) used in prior work for strongly supervised moment retrieval task (*e.g.*, [14]), but the PE features provide information about the temporal position of each frame rather than the approximate position index at the segment level. This helps to circumvent the lack of recurrent networks in our approach. For the desired PE features of dimension d, let pos indicates the temporal position of each frame, i is the index of the feature being encoded, and M is a scalar constant, then the PE features are defined as:

$$PE_{pos,i} = \begin{cases} \sin(pos/M^{i/d}) & \text{if } i \text{ is even} \\ \cos(pos/M^{i/d}) & \text{otherwise.} \end{cases} \quad (1)$$

We found that $M = 10{,}000$ works well for all videos. These PE features are concatenated with the frame features at corresponding frame position before going to the cross-modal interaction layers.

3.1.1 Frame-By-Word Interaction

Rather than relating a sentence-level representation with each frame as done in prior work [25, 20], we aggregate similarity scores between all frame and word combinations from the input video and sentence. Intuitively, words and frames that are semantically similar should have higher scores. These Frame-By-Word (FBW) similarity scores are used to compute attention weights that identify important frame and word combinations for retrieving the correct

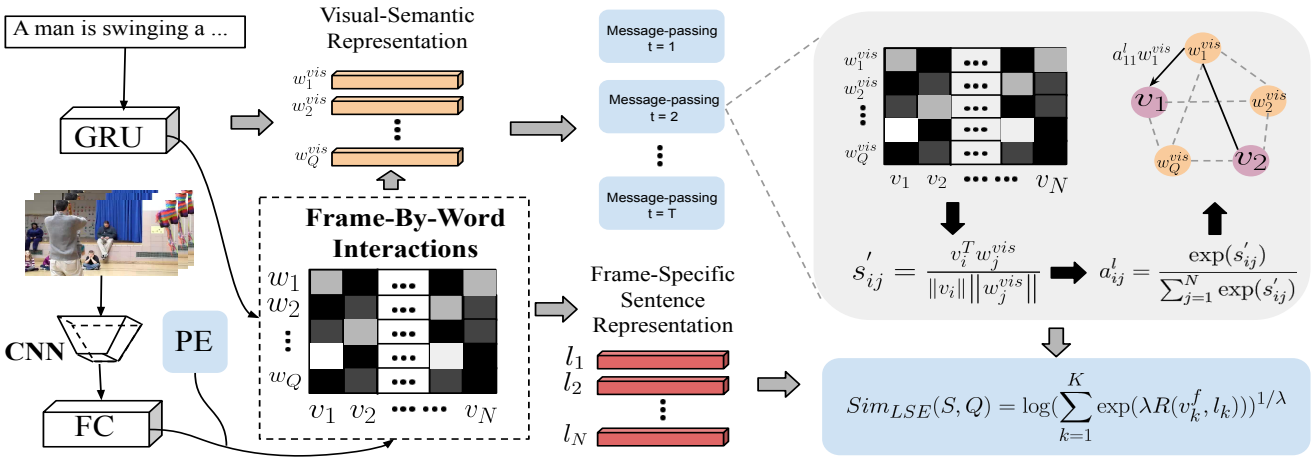

Figure 2. An overview our LoGAN model. Query words are encoded by GRU and the outputs at each time step are used as our word representations. Frames are represented by the output of a pretrained CNN followed by a fully-connected (FC) layer. The visual representations of each frame are concatenated with positional encodings to integrate their relative positions in the video sequence. Our whole model consists of a two-stage multimodal interaction mechanism - Frame-By-Word Interactions (Section 3.1.1) and the WCVG (Section 3.2). The Frame-By-Word interactions are formulated under the assumption that frames and words that correspond to each other should have higher similarity scores and vice versa. Without recurrence, WCVG updates each node (frame) with visual and semantic information from other nodes through a series of message-passing iterations.

video segment. More formally, for N video frames and Q words in the input, we compute:

$$s_{ij} = \frac{v_i^T w_j}{\|v_i\| \, \|w_j\|} \text{ where } i \in [1, N], j \in [1, Q]. \quad (2)$$

Note that v now represents the concatenation of the video frame features and the PE features.

Frame-Specific Sentence Representations. We obtain the normalized relevance of each word w.r.t. to each frame from the FBW similarity matrix, and use it to compute attention for each word:

$$a_{ij} = \frac{\exp(s_{ij})}{\sum_{j=1}^{Q} \exp(s_{ij})}. \quad (3)$$

Using the above-mentioned attention weights, a weighted combination of all the words are created, with correlated words to the frame gaining high attention. Intuitively, a word-frame pair should have a high similarity score if the frame contains a reference to the word. Then the frame-specific sentence representation emphasizes words relevant to the frame and is defined as:

$$l_i = \sum_{j=1}^{Q} a_{ij} w_j. \quad (4)$$

Note that these frame-specific sentence representations don't participate in the iterative message-passing process (Section 3.2). Instead, they are used to infer the final similarity score between a video segment and the query (Section 3.3).

Word-Specific Video Representations. We compute attention weights of each frame using the normalized relevance of each frame w.r.t. each word:

$$a'_{ij} = \frac{\exp(s_{ij})}{\sum_{i=1}^{N} \exp(s_{ij})}. \quad (5)$$

We use these attention weights to create a weighted combination of visual frame features determined by the relevance of each frame to the word. *I.e.*:

$$f_j = \sum_{i=1}^{N} a'_{ij} v_i. \quad (6)$$

These word-specific video representations are used in our Word-Conditioned Visual Graph, which we will discuss in the next section.

3.2. Word-Conditioned Visual Graph

Given the sets of visual representations, word representations and their corresponding word-specific video representations, WCVG aims to learn contextualized visual-semantic representations by integrating temporal contextual information into the visual features. Instead of simply modeling relational context between video frames using a recurrent network as done in prior work [3], WCVG seeks to model the relationships between all possible pairs of frames. This is based on our obervation that a video frame can be related to other frames in many ways given different contexts, as described by the sentence. We conjecture that an LSTM may not be able to capture long-range dependencies, which

is the case with longer videos such as in Charades-STA. In contrast, graph networks have been shown to facilitate learning contextualized video representations by aggregating information from other nodes through message-passing.

To begin, the word representations w_j are concatenated with its word-specific video representation f_j to create a new visual-semantic representation w_j^{vis}. Intuitively, the visual-semantic representations not only contain the semantic context of each word but also a summary of the video with respect to each word. A complete bipartite graph is constructed to connect all nodes containing visual features v_i to all nodes containing visual-semantic features w_j^{vis}.

Iterative Word-Conditioned Message-Passing. The iterative message-passing process introduces a second round of FBW interaction, similar to that in Section 3.1.1, to infer the latent temporal correspondence between each frame v_i and visual-semantic representation w_j^{vis}. The goal is to update the representation of each frame v_i with the video context information from each word-specific video representation w_j^{vis} . To realize this, we first learn a projection W_1 followed by a ReLU of w_j^{vis} to obtain a new word representation to compute a new similarity matrix s'_{ij} on every message-passing iteration, namely, we obtain a replacement for w_j in Eq. (2) via $w'_j = ReLU(W_1(w_j^{vis}))$.

Updates of Visual Representations. We aggregate messages passed from each visual-semantic node w_j^{vis} to each visual node v_i and use them to update the visual nodes. More formally, representations of visual nodes at the t-th iteration are updated by summing incoming messages via:

$$v_i^t = W_2(concat\{v_i^{t-1}; \sum_{j=1}^{Q} a_{ij}^l w'_j\}), \tag{7}$$

where a_{ij} is obtained by applying Eq. (3) to the newly computed FBW similarity matrix s'_{ij}, and W_2 is a learned projection to make v_i^t the same dimensions as the frame-specific sentence representation l_i (Eq. (4)) which are used to compute a sentence-segment similarity score.

3.3. Multimodal Similarity Inference

The final updated visual representations $V^T = \{v_1^T, v_2^T, \cdots, v_V^T\})$ are used to compute the relevance of each frame to its attended sentence-representations. A segment is defined as any arbitrary continuous sequence of visual features. We denote a segment as $S = \{v_1^T, \cdots, v_K^T\}$ where K is the number of frame features contained within the segment S. We adopt the LogSumExp (LSE) pooling similarity metric used in SCAN [19], to determine the relevance each proposal segment to the query:

$$Sim_{LSE}(S,Q) = \log(\sum_{k=1}^{K} \exp(\lambda R(v_k^f, l_k)))^{1/\lambda}$$
$$\text{where } R(v_k, l_k) = \frac{{v_k}^T l_k}{\|v_k\| \, \|l_k\|}. \tag{8}$$

λ is a hyperparameter that weighs the relevance of the most salient parts of the video segment to the corresponding frame-specific sentence representations. We treat each video as a single segment during training, and consider any description from a different video as a negative example. Given a triplet (X^+, Y^+, Y^-), where (X^+, Y^+) is a positive video-sentence pair and (X^+, Y^-) a negative video-sentence pair, we use a margin-based ranking loss L_T to train our model. Our model's total loss is then defined as:

$$L_{total} = \sum_{(V^+,Q^+)} \{\sum_{Q^-} L_T(V^+, Q^+, Q^-) + \sum_{V^-} L_T(Q^+, V^+, V^-)\}. \tag{9}$$

Sim_{LSE} is used to compute similarity for all pairs. Following [35], during training we enumerate all possible triplets in a minibatch and use top-K triplets with the highest loss from Eq.(9) to update our model. The value of K is determined empirically using the validation set. At test time, Sim_{LSE} is also used to rank video segments.

4. Experiments

We evaluate LoGAN on the two datasets below:

Charades-STA [10] is built upon the original Charades [30] dataset which contains video-level paragraph descriptions and temporal annotations for activities. Charades-STA is created by breaking down the paragraphs to generate sentence-level annotations and aligning the sentences with corresponding video segments. In total, it contains 12,408 and 3,720 query-moment pairs in the train and test sets respectively. For fair comparison with the weakly-supervised model TGA [25], we use the same non-overlapping sliding windows with the temporal length of 128 and 256 frames to generate candidate temporal segments.

DiDeMo [14] consists of 8395/1004/1065 train/test/validation videos collected from Flickr. Each query contains the temporal annotations from at least 4 different annotators. Each video is limited to a maximum duration of 30 seconds and equally divided into six segments with five seconds each. With the five-second segment as basic temporal unit, there are 21 possible candidate temporal segments for each video. These 21 segments will used to compute the similarities with the input query and the top scored segment will be returned as the localization result.

Method	Training Supervision	iou = 0.3			iou = 0.5			iou = 0.7		
		R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
(a) CTRL [10]	Strong	-	-	-	23.63	58.92	-	8.89	29.52	-
MLVI [38]	Strong	54.7	95.6	99.2	35.6	79.4	93.9	15.8	45.4	62.2
DEBUG [22]	Strong	54.95	–	–	37.39	–	–	17.69	–	–
MAN [40]	Strong	-	-	-	46.53	86.23	-	22.72	53.72	-
(b) TGA [25]	Weak	29.68	83.87	98.41	17.04	58.17	83.44	6.93	26.80	44.06
SCN [20]	Weak	42.96	**95.56**	-	23.58	71.80	-	9.97	38.87	-
LoGAN (ours)	Weak	**51.67**	92.74	**99.46**	**34.68**	**74.30**	**86.59**	**14.54**	**39.11**	**45.24**
Upper Bound	-	-	-	99.84	-	-	88.17	-	-	46.80

Table 1. Moment retrieval performance comparison on the Charades-STA test set. (a) contains representative results of fully-supervised methods reported in prior works while (b) compares weakly-supervised methods. Our approach outperforms prior work by a significant margin.

Method	#Params	Vision-Language Interaction			iou = 0.3		iou = 0.5		iou = 0.7		rSUM
		Video	Text	Co-attention?	R@1	R@5	R@1	R@5	R@1	R@5	
TGA [25]	3M	Frame	Sentence	✗	29.68	83.87	17.04	58.17	6.93	26.80	222.49
TGA [25]	19M	Frame	Sentence	✗	27.36	77.58	14.38	59.97	5.24	30.40	214.93
SCN [20]	–	Frame	Sentence	✗	42.96	**95.56**	23.58	71.80	9.97	38.87	282.74
LCGN [15]	152M	Frame	Word & Sentence	✓	35.81	82.93	19.25	65.11	7.12	32.90	124.38
MAN [40]	11M	Segment	Word	✓	13.60	69.30	5.94	46.05	1.37	21.51	157.77
FBW	3M	Frame	Word	✓	38.13	90.59	24.73	69.92	9.73	34.20	267.30
FBW	20M	Frame	Word	✓	38.73	91.10	24.71	69.19	10.11	33.17	267.01
LoGAN (ours)	11M	Frame	Word	✓	**51.67**	92.74	**34.68**	**74.30**	**14.54**	**39.11**	**307.04**

Table 2. Comparison of the different types of multimodal interaction components including the types of features used in different models and their performance on Charades-STA test set. Note that the number of parameters for SCN is not reported because the code is not released.

Method	Training Supervision	R@1	R@5	mIOU
(a) MCN	Strong	28.10	78.21	41.08
TGN	Strong	28.23	79.26	42.97
(b) TGA	Weak	12.19	39.74	24.92
LoGAN	Weak	**39.20**	**64.04**	**38.28**
Upper Bound	-	74.75	100.00	96.05

Table 3. Moment retrieval comparison on the DiDeMo test set. (a) contains fully-supervised MCN [14] and TGN [3] results reported in prior work. (b) compares weakly-supervised TGA [25] and our approach. Similar to our results on Charades-Sta, LoGAN obtains a significant improvement over prior work as well, demonstrating its versatility to working with video frames or segments.

Metrics. On the DiDeMo dataset, we adopt the mean temporal Intersection-Over-Union (tIOU) and Recall@N (R@N) at IOU threshold = θ, or percent of times at least one segment in the top N has a tIOU of at least θ. We also report mIOU, which is the average IOU with the ground-truth segments for the highest ranking segment to each query input. On the Charades-STA dataset, only the R@N metric is used for evaluation.

Implementation Details. For fair comparison, we utilize the same input features as [25]. Specifically, the word representations are initialized with GloVe [27] and fine-tuned during training. DiDeMo [14] provides mean-pooled optical flow and VGG [31] features that we make use of. For Charades-STA we compute C3D [33] features. We adopt an initial learning rate of $1e^{-5}$ and a margin$= 0.7$ used in our model's triplet loss (Eq. 9). In addition, we use three iterations for the message-passing process. On both datasets, the output dimensions of all layers are set to 512. During training time, we sample the top 15 highest-scoring negative videos as negative samples. Our model is trained end-to-end using the ADAM [18] optimizer.

4.1. Comparison to Prior Work

Table 1 compares our LoGAN model to results reported in prior work. Notably, our Recall@1 outperforms the state-of-the-art weakly-supervised approach SCN by 5-11%, while also boasting better performance in most other metrics. This is especially significant since SCN uses a Transformer architecture [34] that has performed so well across many vision-language tasks in strongly-supervised settings [23, 32]. We also get about the same or better

Dataset	Charades-STA							DiDeMo			
	iou = 0.3		iou = 0.5		iou = 0.7						
Method	R@1	R@5	R@1	R@5	R@1	R@5	rSUM	R@1	R@5	mIOU	SUM
(a) LoGAN Ablation Study											
FBW	41.41	93.79	26.91	72.04	10.71	35.09	279.95	33.02	66.29	38.37	137.68
FBW + PE	36.34	**94.40**	29.02	71.52	12.69	36.22	280.19	39.93	66.53	39.19	145.65
FBW-WCVG	43.99	90.85	28.85	71.76	11.40	35.58	282.43	39.59	61.14	39.27	140.00
FBW-WCVG + TEF	43.99	88.03	28.01	69.19	11.20	35.29	275.71	37.55	66.36	39.11	143.02
FBW-WCVG *	42.18	91.09	27.92	71.43	11.24	35.69	279.55	39.06	66.31	39.05	144.42
FBW-WCVG + PE (LoGAN)	**46.05**	91.25	**30.09**	**73.49**	**13.70**	**38.32**	**292.90**	**41.62**	**66.57**	**39.20**	**147.39**
(b) # Message-Passing Iterations											
LoGAN (2)	43.39	86.14	15.18	68.89	13.10	36.14	262.84	40.21	66.72	39.14	146.07
LoGAN (3)	**46.05**	**91.25**	**30.09**	**73.49**	**13.70**	**38.32**	**292.90**	**41.62**	**66.57**	**39.20**	**147.39**
LoGAN (4)	43.71	88.07	15.31	68.94	13.10	36.50	265.63	40.01	66.16	39.06	145.23

Table 4. Results from the validation sets of both datasets. (a) contains ablations of the components of LoGAN while (b) reports the effect the number of message-passing iterations has on performance. *indicates the same number of model parameters as the full LoGAN model. Notably, WCVG has been shown to improve retrieval accuracy significantly when used in conjunction with positional embeddings across both datasets. In particular, we report large improvement in the R@1 accuracy, which is the hardest metric. As with other graph convolutional networks, the number of message-passing iterations is often a critical hyperparameter that has to be carefully tuned for best performance. In our experiments, 3 rounds of message-passing yields the best performance over both dataset.

performance than several of the strongly-supervised methods, which as we note earlier also uses a co-attention module that is also based on frame-by-word interactions like ours, but doesn't have our WCVG to learn relationships between frames or positional encodings. In particular, LoGAN achieves results that are almost on par with those of DEBUG, which is a strongly-supervised transformer-based method. Transformers have been shown to be state-of-the-art in computing self-attention over sequences.

Following TGA, the upper bounds are computed from picking the sliding window proposals with the highest overlap with the ground-truth segments. Note that R@10 performance of our LoGAN model in Table 1 is nearly at the upper bound performance. Thus, in Table 2, where we compare different ways of learning correlations between the video and text features, we only report R@{1,5} and then use the sum of recalls (rSUM) at different thresholds to rank methods. Our FBW interaction module, which does not use our PE features or update features using WCVG, gets nearly the same performance as the full SCN model, the best performing method from all compared methods including other weakly-supervised methods and adaptations from related work.

Of course, our full LoGAN model outperforms SCN by 18 points, demonstrating a clear benefit over prior work. The adaptations we compare to include MAN, that uses a co-attention module between words and segments developed for the strongly-supervised setting, and the LCGN model that was initially designed for use on images to perform VQA. We outperform both methods, even though LCGN also uses in order of magnitude more model parameters. This helps confirm our claim that methods designed for other tasks/settings do not always generalize to the weakly-supervised moment retrieval task.

Table 3 compares our approach with prior work on the DiDeMo dataset. We obtain a 13-27 point improvement over the state-of-the-art weakly-supervised approaches on this dataset. Even more significant is the 11% gain on R@1 over the fully supervised TGN model, which as we note in the introduction uses a recurrent network to learn interactions between sentences and frames using a recurrent network rather than directly computing co-attention from the features. This suggest that recurrent networks find it difficult to capture temporal dependencies, which we avoid by jointly reasoning over the entire video.

4.2. Analyzing LoGAN

Table 4(a) provides an ablation study of the components of LoGAN on both datasets. Note that our combined LoGAN model is comprised of the FBW and WCVG components as well as the incorporation of PEs. The results obtained by our FBW variant demonstrate that capturing fine-grained frame-by-word interactions is essential to infer the latent temporal alignment between these two modalities. The trends on both datasets are largely consistent, with the PE features providing 3-8 point overall improvement over the FBW interaction module and using the WCVG module to jointly reason over the entire video also provides a 3-8 point overall improvement. Notably, the PE features outperform the temporal endpoint features (TEF), which boosted performance in the strongly-supervised setting in prior work [14]. We observe that TEFs actually

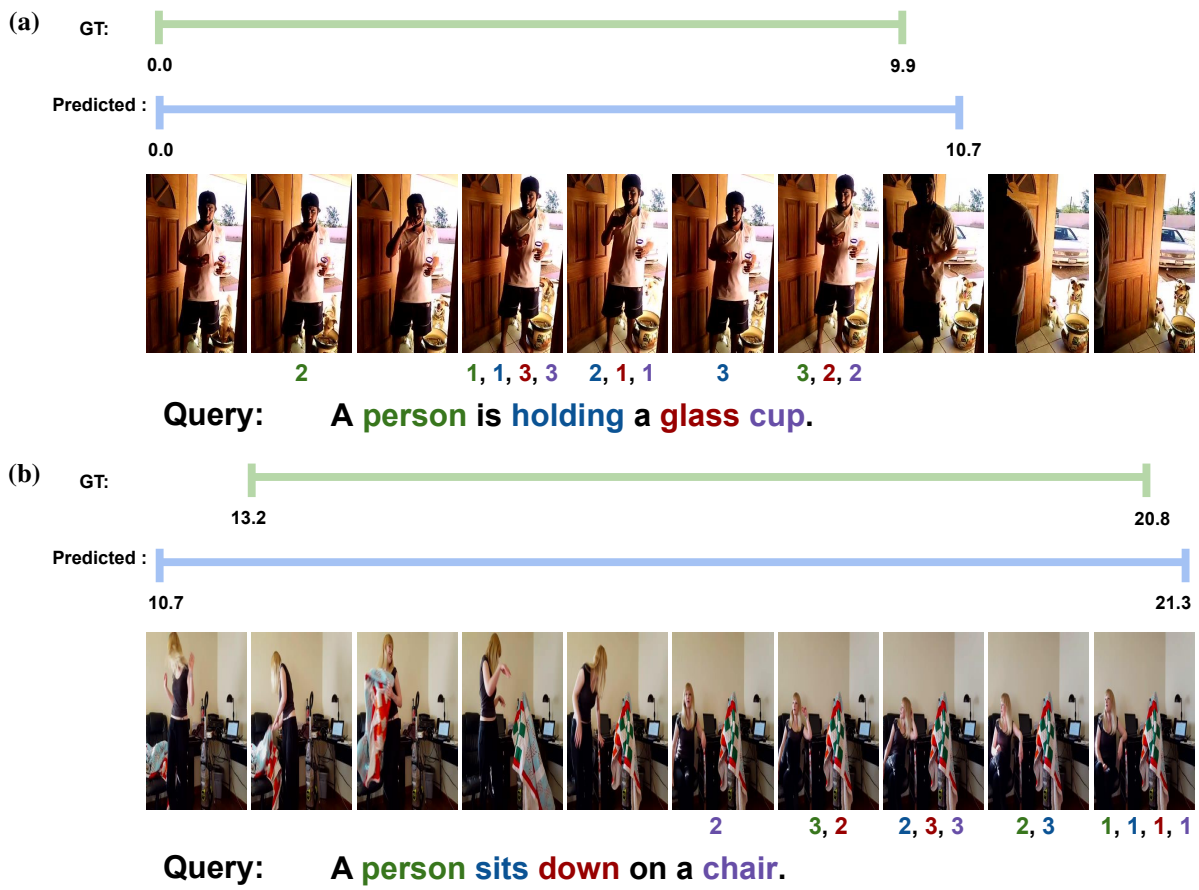

Figure 3. Visualization of the top three highest weights assigned to the frames for each word. Similar colors indicate correspondence. As shown in these examples, frames that are highly correlated with the query words generally fall into the GT temporal segment. Moreover, the frames with the largest attention weights with respect to each word are distributed over the same subset of frames.

hurt performance slightly. We theorize that the positional encodings aid in integrating temporal context and relative positions into the learned visual-semantic representations. This makes it particularly useful for Charades-STA since its videos are generally much longer.

Table 4(b) reports how performance changes over different numbers of message passing iterations using WCVG, finding that performance saturates after three rounds. It is interesting that using 4 iterations reduces performance instead. We hypothesize that using 4 iterations dilutes the information contained in the initial visual frame representations, which are still essential for inference.

We provide qualitative results in Figure 3 to provide further insights into our model. In both examples, we observe that the top three salient frames with respect to each word generally distribute over the same subset of frames, which is actually the ground truth temporal segment. This suggests that our proposed model is able to identify the most salient frames with respect to each word, which further helps the temporal localization in our task.

5. Conclusion

In this work, we propose our Latent Graph Co-Attention Network which leverages fine-grained frame-by-word interactions to model relational context between all possible pairs of video frames given the semantics of the query. Learning contextualized visual-semantic representations helps our model to reason more effectively about the temporal occurrence of an event as well as the relationships of entities described in the natural language query. Our experimental results empirically demonstrate the effectiveness of such representations on the accurate localization of video moments. Finally, our work also provides a useful reference for future work in video moment retrieval and latent multimodal reasoning in video-and-language tasks.

Acknowledgements: This work is supported in part by DARPA and NSF awards IIS-1724237, CNS-1629700, CCF-1723379.

References

[1] Hedi Ben-Younes, Rémi Cadene, Matthieu Cord, and Nicolas Thome. Mutan: Multimodal tucker fusion for visual question answering. In *Proceedings of the IEEE international conference on computer vision*, pages 2612–2620, 2017.

[2] Remi Cadene, Hedi Ben-Younes, Matthieu Cord, and Nicolas Thome. Murel: Multimodal relational reasoning for visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1989–1998, 2019.

[3] Jingyuan Chen, Xinpeng Chen, Lin Ma, Zequn Jie, and Tat-Seng Chua. Temporally grounding natural sentence in video. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 162–171, 2018.

[4] Jingyuan Chen, Lin Ma, Xinpeng Chen, Zequn Jie, and Jiebo Luo. Localizing natural language in videos. In *AAAI Conference on Artificial Intelligence*, 2019.

[5] Kan Chen, Rama Kovvuri, and Ram Nevatia. Query-guided regression network with context policy for phrase grounding. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 824–832, 2017.

[6] Shaoxiang Chen and Yu-Gang Jiang. Semantic proposal for activity localization in videos via sentence query. In *AAAI Conference on Artificial Intelligence*, 2019.

[7] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*, 2014.

[8] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[9] Fartash Faghri, David J Fleet, Jamie Ryan Kiros, and Sanja Fidler. Vse++: Improving visual-semantic embeddings with hard negatives. In *Proceedings of the British Machine Vision Conference (BMVC)*, 2018.

[10] Jiyang Gao, Chen Sun, Zhenheng Yang, and Ram Nevatia. Tall: Temporal activity localization via language query. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5267–5275, 2017.

[11] Mingfei Gao, Larry S Davis, Richard Socher, and Caiming Xiong. Wslln: Weakly supervised natural language localization networks. *arXiv preprint arXiv:1909.00239*, 2019.

[12] Runzhou Ge, Jiyang Gao, Kan Chen, and Ram Nevatia. Mac: Mining activity concepts for language-based temporal localization. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 245–253. IEEE, 2019.

[13] Soham Ghosh, Anuva Agarwal, Zarana Parekh, and Alexander Hauptmann. Excl: Extractive clip localization using natural language descriptions. *arXiv preprint arXiv:1904.02755*, 2019.

[14] Lisa Anne Hendricks, Oliver Wang, Eli Shechtman, Josef Sivic, Trevor Darrell, and Bryan Russell. Localizing moments in video with natural language. In *International Conference on Computer Vision (ICCV)*, 2017.

[15] Ronghang Hu, Anna Rohrbach, Trevor Darrell, and Kate Saenko. Language-conditioned graph networks for relational reasoning. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2019.

[16] Andrej Karpathy and Li Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3128–3137, 2015.

[17] Jin-Hwa Kim, Jaehyun Jun, and Byoung-Tak Zhang. Bilinear attention networks. In *Advances in Neural Information Processing Systems*, pages 1564–1574, 2018.

[18] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference for Learning Representations*, 2015.

[19] Kuang-Huei Lee, Xi Chen, Gang Hua, Houdong Hu, and Xiaodong He. Stacked cross attention for image-text matching. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 201–216, 2018.

[20] Zhijie Lin, Zhou Zhao, Zhu Zhang, Qi Wang, and Huasheng Liu. Weakly-supervised video moment retrieval via semantic completion network. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020.

[21] Jingyu Liu, Liang Wang, and Ming-Hsuan Yang. Referring expression generation and comprehension via attributes. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4856–4864, 2017.

[22] Chujie Lu, Long Chen, Chilie Tan, Xiaolin Li, and Jun Xiao. DEBUG: A dense bottom-up grounding approach for natural language video localization. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 2019.

[23] Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. *arXiv preprint arXiv:1908.02265*, 2019.

[24] Jiasen Lu, Jianwei Yang, Dhruv Batra, and Devi Parikh. Hierarchical question-image co-attention for visual question answering. In *Advances in neural information processing systems*, pages 289–297, 2016.

[25] Niluthpol Chowdhury Mithun, Sujoy Paul, and Amit K Roy-Chowdhury. Weakly supervised video moment retrieval from text queries. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 11592–11601, 2019.

[26] Hyeonseob Nam, Jung-Woo Ha, and Jeonghee Kim. Dual attention networks for multimodal reasoning and matching. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 299–307, 2017.

[27] Jeffrey Pennington, Richard Socher, and Christopher Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014.

[28] Bryan A Plummer, Paige Kordas, M Hadi Kiapour, Shuai Zheng, Robinson Piramuthu, and Svetlana Lazebnik. Conditional image-text embedding networks. In *Proceedings of the*

European Conference on Computer Vision (ECCV), pages 249–264, 2018.

[29] Zheng Shou, Dongang Wang, and Shih-Fu Chang. Temporal action localization in untrimmed videos via multi-stage cnns. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1049–1058, 2016.

[30] Gunnar A. Sigurdsson, Gül Varol, Xiaolong Wang, Ali Farhadi, Ivan Laptev, and Abhinav Gupta. Hollywood in homes: Crowdsourcing data collection for activity understanding. In *European Conference on Computer Vision*, 2016.

[31] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[32] Chen Sun, Austin Myers, Carl Vondrick, Kevin Murphy, and Cordelia Schmid. Videobert: A joint model for video and language representation learning. *arXiv preprint arXiv:1904.01766*, 2019.

[33] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 4489–4497, 2015.

[34] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

[35] Liwei Wang, Yin Li, Jing Huang, and Svetlana Lazebnik. Learning two-branch neural networks for image-text matching tasks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(2):394–407, 2018.

[36] Yu Wu, Linchao Zhu, Yan Yan, and Yi Yang. Dual attention matching for audio-visual event localization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.

[37] Huijuan Xu, Abir Das, and Kate Saenko. R-c3d: Region convolutional 3d network for temporal activity detection. In *Proceedings of the IEEE international conference on computer vision*, pages 5783–5792, 2017.

[38] Huijuan Xu, Kun He, Bryan A Plummer, Leonid Sigal, Stan Sclaroff, and Kate Saenko. Multilevel language and vision integration for text-to-clip retrieval. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 9062–9069, 2019.

[39] Yitian Yuan, Tao Mei, and Wenwu 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*, volume 33, pages 9159–9166, 2019.

[40] Da Zhang, Xiyang Dai, Xin Wang, Yuan-Fang Wang, and Larry 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*, pages 1247–1257, 2019.

[41] Yue Zhao, Yuanjun Xiong, Limin Wang, Zhirong Wu, Xiaoou Tang, and Dahua Lin. Temporal action detection with structured segment networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2914–2923, 2017.