Hierarchical Attention Based Spatial-Temporal Graph-to-Sequence Learning for Grounded Video Description

Kai Shen[1*], **Lingfei Wu**[2*], **Fangli Xu**[3], **Siliang Tang**[1†], **Jun Xiao**[1] and **Yueting Zhuang**[1]
[1]Zhejiang University
[2]IBM Research
[3]Squirrel AI Learning
{shenkai,siliang,junx,yzhuang}@zju.edu.cn, wuli@us.ibm.com, lili@yixue.us

Abstract

The task of Grounded Video Description (GVD) is to generate sentences whose objects can be grounded with the bounding boxes in the video frames. Existing works often fail to exploit structural information both in modeling the relationships among the region proposals and in attending them for text generation. To address these issues, we cast the GVD task as a spatial-temporal Graph-to-Sequence learning problem, where we model video frames as spatial-temporal sequence graph in order to better capture implicit structural relationships. In particular, we exploit two ways to construct a sequence graph that captures spatial-temporal correlations among different objects in each frame and further present a novel graph topology refinement technique to discover optimal underlying graph structure. In addition, we also present hierarchical attention mechanism to attend sequence graph in different resolution levels for better generating the sentences. Our extensive experiments demonstrate the effectiveness of our proposed method compared to state-of-the-art methods.

1 Introduction

The task of Grounded video description (GVD) [Zhou *et al.*, 2019] aims to generate more grounded and accurate descriptions by linking the generated words with the regions in video frames. Compared to conventional video description task that generates a human-like sentence to describe the video contents [Zhou *et al.*, 2018], GVD has advantages of modelling the video by objects and associating the generated text with them to describe the video in a high-quality and grounded way.

However, current state-of-the-art GVD methods often fail to exploit structural information both in two aspects: i) modeling the relationships among the region proposals; and ii) attending them for text generation. On one hand, existing works either encode region proposals independently or using self-attention-based mechanisms [Zhou *et al.*, 2019]. Therefore, it either fails to consider implicit structural information among the region proposals or needs to handle noisy or fake relationships among objects. In addition, the explicit structural features of objects (eg. spatial, temporal, semantic) which are potentially important to discover the true correlations among the objects, are overlooked using self-attention only.

On the other hand, when generating sentences, most previous works adopted top-down attention (means the objects are attended equally and individually) to focus on the relevant objects directly, regardless whether the video frames that these objects are located are semantically related in a high level. Although it can reduce the loss of grounding, the structural correlations of the video frames are completely ignored. However, for a specific word generation step, it is more reasonable to focus on a certain segment of the video frames first and then focus on the objects in these frames.

More recently, the graph-based method for video understanding started attracting more attentions in some close related fields such as image caption [Li *et al.*, 2019b]. However, due to the complexity of video understanding, there still remains significant challenges to adapt these graph-based approaches into the GVD task. The first challenge comes from the unique properties of video - how to model the spatial-temporal correlations using a graph. So far, the existing way of building a graph for visual contents, such as the scene graph, only focuses on the single static image [Yang *et al.*, 2019]. The technology that can effectively construct a graph for image sequences like a video is still unclear and worth exploring. Even we can model a video with a graph, another challenge still remains. Due to the temporal redundancy in video frames, similar objects staying in many frames. As a result, the constructed graph can be very noisy since there are many useless edges in the graph. This will mislead the model, and it may learn less discriminative features for downstream tasks such as generating the description. Therefore, the constructed graph structure should be refined according to the downstream tasks.

To address the aforementioned issues, we cast the GVD task as a graph-to-sequence learning problem and propose Hierarchical Attention based Spatial-Temporal Graph-to-Sequence Learning framework (HAST-Graph2Seq) for Grounded Video Description. Specifically, we introduce spatial-temporal sequence graph $\mathbf{A}$ to capture the implicit correlations among region proposals, whose topology is ini-

[*]Both Authors Contributed Equally
[†]Corresponding Author

tially obtained in pre-processing with or without external knowledge. Furthermore, we train a similarity metric to construct a semantically implicit graph $\mathbf{A}_{implicit}$ to refine the noisy initial graph $\mathbf{A}_{init}$ through an end-to-end training for learning node (object) embeddings via graph neural networks. For the decoding procedure, we introduce hierarchical graph attention on the refined sequence-graph for description generation by first finding the regions of frames by attending a certain segment of the given video frames and then finding the regions of objects located in these related frames.

In summary, we highlight our main contributions below:

- We cast the GVD task as a spatial-temporal Graph-to-Sequence learning problem, where we model video frames as sequence graph to better capture implicit spatial-temporal structural relationships. To the best of our knowledge, this is the first time a spatial-temporal Graph-to-Sequence model is presented for GVD task.
- In particular, we exploit two ways to construct a sequence graph that captures spatial-temporal correlations among different objects in each frames and further present a novel graph topology refinement techniques to discover optimal underlying graph structure.
- We also present hierarchical attention mechanism to attend sequence graph in different resolution levels for better generating the sentences. The results demonstrate the effectiveness of our proposed method.

2 Related Work

2.1 Visual Description

With the rapid development of deep learning in CV and NLP, video description begins to generate the description of a video using the attention-based encoder-decoder like architectures [Venugopalan *et al.*, 2015; Xu *et al.*, 2018b; Liu *et al.*, 2016]. These methods are effective but they overlook the fine-grained object clues that separated in frames. Borrowing the ideas of spatial-attention in image caption domain [Anderson *et al.*, 2018; Liu *et al.*, 2018; Li *et al.*, 2019a], many works model the video in both global video features and regional object features. In [Zhou *et al.*, 2019], they encode the objects with transformer[Vaswani *et al.*, 2017] and then link the words of generated descriptions with the clues in certain regions in the video to generate descriptions more grounded. However, since not all objects are key for generating and not all objects have much to do with others, the methods like self-attention may confuse the model. Therefore, graph-based methods which model the regions with abundant semantic relations are introduced to this area. [Yao *et al.*, 2018] uses relation prediction methods to generate a semantic graph to explore visual relations. [Yang *et al.*, 2019] uses scene graphs that have richer relation clues in image caption. Further, [Zhang and Peng, 2019] constructs the bi-directional temporal trajectory graph based on similarity and attend on them hierarchically. Although these methods achieve great success, they are constructed by link prediction methods independent of the description generation task so they are noisy to it. [Wang and Gupta, 2018] defines the graph topology based on learned similarity. In [Tomei *et al.*, 2019], they employ a self-attention based semantic graph to construct the topology. Although they consider the implicit relations of the objects, the semantic graphs are handling individually. Due to the hardness of video understanding, it is hard to build a proper topology by attention thoroughly.

2.2 Graph-to-sequence Learning

Graph-to-sequence learning has been surge of interests recently in the NLP domain. The main goal for graph-to-sequence learning is to generate sequential content from graph structured data, which learns a mapping between graph inputs to sequence outputs through attention-based mechanisms [Xu *et al.*, 2018a; Chen *et al.*, 2020; Gao *et al.*, 2019]. However, since there is no explicit graph structure for video, it is hard to adapt these methods directly.

Unlike these previous methods, we propose a novel Hierarchical Attention based Spatial-Temporal Graph-to-Sequence Learning framework considering both the modeling and the usage of regions in encoder and decoder, including initial graph construction, noisy initial topology refinement and attending on the graph hierarchically.

3 HAST-Graph2Seq Framework for GVD

The GVD task aims to generate a text description $\mathbf{S}_{gt}$ from a video segment denoted as $\mathbf{V}$. In training stage, we will uniformly sample F frames from each video segment as $\mathbf{V}_{sample} = \{\mathbf{v}_1, \mathbf{v}_2, ..., \mathbf{v}_F\}$, and provide N_{gt} object regions in $\mathbf{V}_{sample}$ which are corresponding to words in $\mathbf{S}_{gt}$. But object regions will not be given in the inference stage.

To make the statement clear, we will give the mathematical notations of the concept mentioned. The video segment is denoted as $\mathbf{v} = \{\mathbf{v}_i\}_{i=1}^{n} \in \mathbf{V}$. The target sentence is $\mathbf{s} = \{\mathbf{s}_i\}_{i=1}^{m} \in \mathbf{S}$, m is the length of the sentence. And we define N_f object regions of each sampled frames in $\mathbf{V}_{sample}$ which are denoted as $\mathbf{R} = \{\mathbf{R}_1, \mathbf{R}_2, ..., \mathbf{R}_F\} = \{\mathbf{r}_1, \mathbf{r}_2, ..., \mathbf{r}_N\} \in \mathbb{R}^{d \times N}$, where d is the dimension of the proposals and $N = \sum_{f=1}^{F} N_f$ is the amounts of the proposals.

As Figure 1 illustrates, we encode the video in two streams. Firstly, we encode the global video features in the Video Global Encoder (Figure 1 a). Then we encode the regions by spatial-temporal sequence graph whose topology will be refined in Graph with Refinement Encoder (Figure 1 b). Finally, we adapt top-down attention by applying temporal attention to global video features and hierarchical graph attention on the spatial-temporal sequence graph in the Language Decoder (Figure 1 c).

3.1 Video Global Encoder

We model the video's global level feature by a Bi-directional LSTM network like most works[Zhou *et al.*, 2019] given by: $\mathbf{h} = BiLSTM(\mathbf{v}) = \{\mathbf{h}_1, \mathbf{h}_2, ..., \mathbf{h}_m\}$ where $\mathbf{v} \in \mathbb{R}^{n \times d}$ is the global feature extracted by a pre-trained 3D-ConvNet [Tran *et al.*, 2015].

3.2 Graph with Refinement Encoder

In this section, we propose a novel visual representation method from the perspective of regions. First of all, inspired

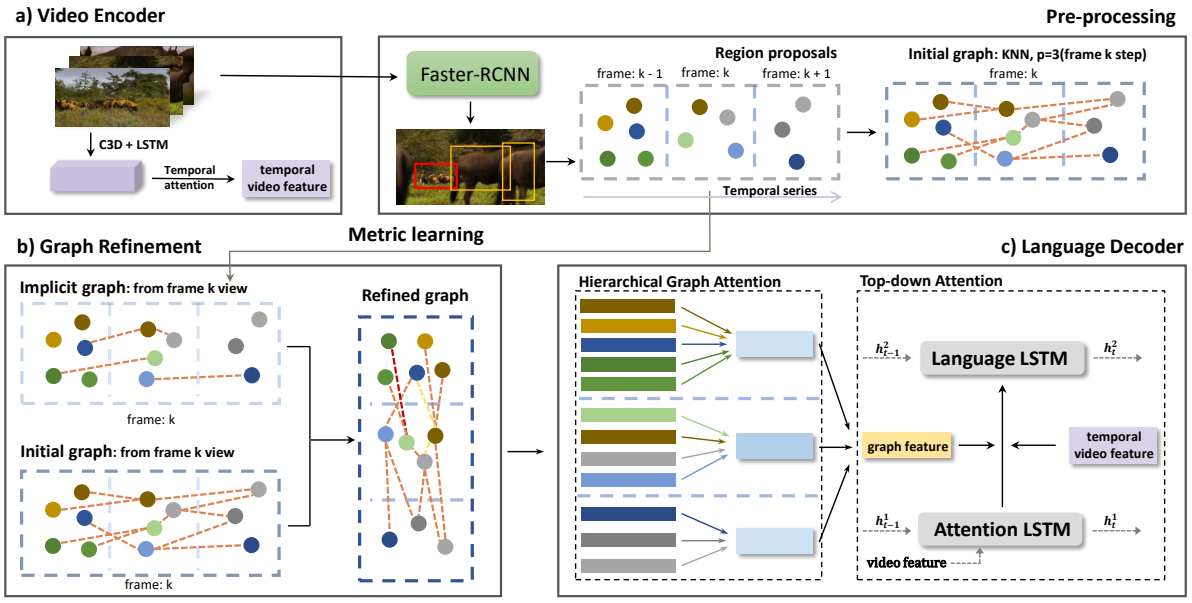

Figure 1: Overall framework of our HAST-Graph2Seq: (a) The video encoder. (b) The graph refinement module. (c) The language module.

by [Zhou *et al.*, 2019], we enhance the proposal features by adding the position and class features. As for proposal modeling, we propose a novel spatial-temporal sequence graph data structure, whose initial topology is obtained before training and refined in end-to-end manner. Notably, the initial topology can be obtained with or without prior knowledge considering the generality.

Feature Enhancement

In this part, we follow [Zhou *et al.*, 2019]'s work, which fusing the spatial-temporal and class features with the original features to enrich them.

(1) For each proposal, we define its' spatial and temporal information as a 5-D list, 4 values for normalized spatial location and 1 value for the normalized frame index. Then we project it to a d_{sp} dimension space. So, the spatial-temporal features of region proposals are denoted as $\mathbf{M}_{sp}$.

(2) We assume that each region proposal $\mathbf{r}_i$ has a class label $c_i \in \{c_1, c_2, ..., c_k\}$. We transfer detection model's weight which is pre-trained on VG dataset to initialize the class embedding denoted as $\mathbf{W}_c \in \mathbb{R}^{d\times k}$ and $\mathbf{B}_c \in \mathbb{R}^{1\times k}$, where d is the embedding dimension. Then we use a attention method to assign each region proposal a class representation: $\mathbf{M}_r(r_i) = Softmax(\mathbf{W}_c^T \mathbf{r}_i + \mathbf{B}_c \mathbb{I}^T)$

To sum up, the region feature will be given by:

$$\hat{\mathbf{R}} = \mathbf{W}_p[\mathbf{R}|\mathbf{M}_{sp}|\mathbf{M}_r] \quad (1)$$

where $[\ |\ ]$ denotes row-wise concatenation and $\mathbf{W}_p \in \mathbb{R}^{l\times(d+k+d_{sp})}$ is the embedding weight. Then we will apply feature aggregation on the enhanced feature $\hat{\mathbf{R}}$.

We adopt the same classification loss just as [Zhou *et al.*, 2019] do denoted as L_{cls}.

Spatial-temporal Sequence-Graph Data Structure

Here we will define the data structure for region proposals. We will view each proposal ($\mathbf{r_i}$) as a node (r_i) in the video graph. To make the problem easier, we will assume that the graph holds the following principles.

(1) Instead of modeling it as a fully connected graph, we assume that the graph hold the locality: each node r_i in sampled frame $\mathbf{v}_f$ will only have connection (if exist) with nodes in $\mathbf{v}_{f-1}, \mathbf{v}_f, \mathbf{v}_{f+1}$. Through this operation, we capture the local spatial relations in single frames and the local temporal relations between frames. What's more, we define the nodes in one single frame as a sub-graph, which consists of the whole graph through temporal edges.

(2) For simplification, we assume the final graph topology is undirected and weighted. However, since we introduce the spatial-temporal information into the node feature space, this assumption will not cause excessive loss of the key position and temporal characteristics. And it is weighted because we want the edges between nodes to be more meaningful.

Initial Graph Topology

By the constraints above, there are several potential methods to form a graph. Since they are formed during pre-processing, they may contain noise.

(1) Without external knowledge: KNN. If we have no prior knowledge of the given regions, a way is to find the correlations in feature space. For each node $r_i \in \mathbf{v}_f$, we will find p nodes $\boldsymbol{R_p} = \{r_1, r_2, ..., r_p\} \in \{\boldsymbol{R}_{f-1}, \boldsymbol{R}_f, \boldsymbol{R}_{f+1}\}$ by KNN Algorithm and add edges between r_i and $\boldsymbol{R_p}$.

(2) With external knowledge method: Relation Graph. Since the region features are extracted by a pre-trained model trained on VG [Krishna *et al.*, 2017] dataset, we can train a semantic relation classifier [Li *et al.*, 2019b] on it. We adopt almost the same operation except replacing the KNN step by the classifier to find the related nodes set $\boldsymbol{R_p}$.

Refinement Procedure

The graph $\mathbf{A}_{init}$ obtained above is noisy but can be refined during the training process.

Empirically, a powerful metric of relation should be learned from specific task. Inspired by [Chen *et al.*, 2019; Vaswani *et al.*, 2017], we design a multi-head weighted cosine similarity metric function:

$$\mathbf{A}_{implicit[i,j]} = \frac{1}{m}\sum_{k=1}^{m} cos(\mathbf{w}^k \odot \hat{\mathbf{r}}_i, \mathbf{w}^k \odot \hat{\mathbf{r}}_j) \tag{2}$$

where $\odot$ denotes the Hadamard product, $\mathbf{w} \in \mathbb{R}^m$ is the learnable weights, m is the heads number, $\hat{\mathbf{r}}_i \in \hat{\mathbf{R}}$ and $\mathbf{A}_{implicit}$ is the implicit graph. We assume that by highlighting some specific dimensions of the region features, we can find the implicit relations beneficial to the task.

Here we adopt the same principles as the initial graph. So we drop the connections if they are against principles (1).

After that, we prune the implicit graph by a threshold ϵ, which means selecting the useful relations and drop the unimportant to make the graph sparse.

$$\mathbf{A}_{i,j} = \mathbf{A}(i,j) * \mathbb{I}(\mathbf{A}(i,j) > \epsilon) \tag{3}$$

where $\mathbb{I}$ is the indicator function. Then we fuse the initial graph with the implicit graph as follows:

$$\widetilde{\mathbf{A}}_{dir} = \lambda \hat{\mathbf{A}}_{initial} + (1-\lambda)\hat{\mathbf{A}}_{implicit} \tag{4}$$

where λ is the hyper-parameter to balance the trade-off between the initial graph and the learned implicit graph. The $\hat{\mathbf{A}}_{initial}$ and the $\hat{\mathbf{A}}_{implicit}$ are normalized adjacency matrix of the initial graph and implicit graph. The normalization is defined as: $\hat{\mathbf{A}} = \mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2}$ and $\mathbf{D}$ is the degree matrix.

To make the graph undirected, the final adjacency matrix is given by: $\widetilde{\mathbf{A}} = (\widetilde{\mathbf{A}}_{dir} + \widetilde{\mathbf{A}}_{dir}^T)/2$

Feature Aggregation

We adapt the classic spectral graph convolutional network to aggregate the features of the nodes modeled by topology $\widetilde{\mathbf{A}}$.

Inspired by resnet architecture [He *et al.*, 2016], we propose the basic module of our architecture as follows (the layer normalization and dropout operations are omitted):

$$\mathbf{X}^{out} = (\sigma(\widetilde{\mathbf{A}}\mathbf{X}^{in}\mathbf{W}) + \mathbf{X}^{in})/\sqrt{2} \tag{5}$$

where the $\mathbf{X}^{in}$ is the input ($\hat{\mathbf{R}}$ in Eq.1), $\widetilde{\mathbf{A}}$ denotes the normalized adjacency matrix, $\mathbf{W}$ is the trainable weights and σ is the non-linear activation function. And we will stack k basic modules to explore deep correlations of the graph. We denote the regions after aggregation as $\widetilde{\mathbf{R}}$ for further illustration.

3.3 The Language Decoder

In this section, we adapt the top-down attention language model for description generation. The attention LSTM is used to encode the visual features and the language LSTM is used to generate words. Between these two LSTMs, we attend on the global video features on temporal level and apply hierarchical graph attention on spatial-temporal sequence graph to capture the visual object clues in different grains.

Attention LSTM

At time step t, we will fuse the hidden state in $t-1$ which denoted as $\mathbf{h}_{t-1}$ with the pooled frame features $\mathbf{v}_{pool}$ to generate a new hidden state $\mathbf{h}_t^1 \in \mathbb{R}^r$.

Temporal Attention

Firstly, we will attend the global frame features in a coarse-grained way. When generating a new word, we should pay different weights on different frames. We denote the results as $\mathbf{h}_{frame} \in \mathbb{R}^r$.

Hierarchical Graph Attention

When handling the proposal regions, instead of attending each region equally, we propose hierarchical attention on the sequence-graph to hold the graph structure.

Firstly, we will attend on the sub-graph to capture the general area of the video. The sub-graph $\widetilde{\mathbf{R}}_i$ can be represented by $\widetilde{\mathbf{R}} = \{\widetilde{\mathbf{R}}_1, \widetilde{\mathbf{R}}_2, ..., \widetilde{\mathbf{R}}_F\} = \{\widetilde{\mathbf{r}}_1, \widetilde{\mathbf{r}}_2, ..., \widetilde{\mathbf{r}}_N\}$. And then we apply mean-pooling to get the vector representation of each sub-graph given by: $\widetilde{\mathbf{R}}_i = MeanPooling(\widetilde{\mathbf{r}}_{k:l})$. $\widetilde{\mathbf{r}}_{k:l}$ denotes regions from $\widetilde{\mathbf{r}}_k$ to $\widetilde{\mathbf{r}}_l$ belong to sub-graph $\widetilde{\mathbf{R}}_i$. Thus $\widetilde{\mathbf{R}} \in R^{F\times l}$. Then we execute graph-level attention:

$$\mathbf{M}(\widetilde{\mathbf{R}}_i, \mathbf{h}_t^1) = \mathbf{W}^T tanh(\mathbf{W}_1\widetilde{\mathbf{R}}_i + \mathbf{W}_2\mathbf{h}_t^1) \tag{6}$$

where $\mathbf{W}_1 \in \mathbb{R}^{o\times l}, \mathbf{W}_2 \in \mathbb{R}^{o\times r}$. And $\mathbf{W} \in \mathbb{R}^o$ is a row vector. Then we apply softmax on M, given by:

$$\alpha^i = \frac{exp(\mathbf{M}(\widetilde{\mathbf{R}}_i, \mathbf{h}_t^1))}{\sum_{j=1}^{F} exp(\mathbf{M}(\widetilde{\mathbf{R}}_j, \mathbf{h}_t^1))} \tag{7}$$

Therefore, we can get the results denoted as $\boldsymbol{\alpha} = \{\alpha_1, \alpha_2, ..., \alpha_F\} \in \mathbb{R}^F$

Secondly, we apply attention on each sub-graph parallelly. For each sub-graph $\widetilde{\mathbf{R}}_f \in \{\widetilde{\mathbf{R}}_1, \widetilde{\mathbf{R}}_2, ..., \widetilde{\mathbf{R}}_F\}$, we apply the same operation as in Eq.6 and Eq.7. For each $\widetilde{\mathbf{R}}_f$, we can get a attention score $\boldsymbol{\beta}_f \in \mathbb{R}^{N_f}$. So for all frames, the score are represented as $\boldsymbol{\beta} = \{\boldsymbol{\beta}_1, \boldsymbol{\beta}_2, ..., \boldsymbol{\beta}_F\} \in \mathbb{R}^{F\times N_f}$

Finally, we fuse the regions given by:

$$\mathbf{h}_{attention} = \sum_{i=1}^{F}\alpha_i \sum_{j=1}^{N_i}\beta_{i,j}\widetilde{\mathbf{R}}_{i,j} \tag{8}$$

where $\mathbf{h}_{attention} \in \mathbb{R}^l$. Then we apply linear projection to project it to r dimension space.

We adapt the same attention supervision here on both node (region) level and sub-graph (frame) level. Firstly, we define the region is positive if it has over 0.5 IOU (intersection over union) with any ground-truth bounding box. Then we apply cross-entropy loss on $\boldsymbol{\beta}$. Besides focusing the correct regions, we also want the visual-groundable word to focus on the correct frames. So, we define a frame is positive if it has at least one positive region. Then we apply the same cross-entropy loss on $\boldsymbol{\alpha}$ too.

$$L_{attn}^a = -\sum_{i=1}^{F}\sum_{j=1}^{N_f} I_{i,j} log\beta_{i,j},\ L_{attn}^b = -\sum_{i=1}^{F} J_i log\alpha_i, \tag{9}$$

where $I_{i,j} = 1$ only if this region is positive. $J_i = 1$ only if this sub-graph is positive.

Language LSTM

The language LSTM is adopted to generate the words while absorbing the visual clues given by: $\mathbf{h}_t = LSTM(\mathbf{h}_t^1, \mathbf{h}_{frame} + \mathbf{h}_{attention})$. $\mathbf{h}_t$ is used to generate descriptions. We adopt the same MLE loss as [Zhou *et al.*, 2019] which denoted by L_{sent}.

Finally, the overall loss function consists of four parts:

$$L = L_{sent} + \lambda_a L_{attn}^a + \lambda_b L_{attn}^b + \lambda_c L_{cls} \quad (10)$$

4 Experiment

4.1 Dataset

We conduct our experiments on the Grounded ActivityNet-Entities Dataset [Zhou *et al.*, 2019] for evaluation. It contains 15k video with 158k spatially annotated bounding boxes from 52k video segments.

4.2 Implementation Details

In this section, we introduce some implementation details of our HAST-Graph2Seq method.

Data processing. For a fair comparison, the data processing procedure is the same to [Zhou *et al.*, 2019]. For each video segment in the dataset, we uniformly sample 10 frames. And for each frame, we use a Faster R-CNN [Ren *et al.*, 2015] detector with ResNeXt-101 backbone to detect 100 region proposals and extract the feature. The detector is pre-trained on Visual Genome [Krishna *et al.*, 2017]. Finally, for the video feature, the temporal feature map is a stack of frame-wise appearance and motion features.

Hyperparameter settings. We set the threshold ϵ value in Eq.3 to 0.4, λ_a to 0.04, λ_b to 0.08, λ_c to 0.5. and number of heads m in Eq.2 to 5. The KNN hyper-parameter $p \in \{5, 10, 20, 30, 40\}$ vary in the experiments as a results of model validation. The region proposal feature's original dimension d is 2048, the region proposals' embedding dimension l is 1024, the word embedding size is 512, rnn hidden size r is 1024 and GCN's layer k is 3. The λ in Eq.4 is 0.8.

4.3 Evaluation Criteria

To measure the performance of our HAST-Graph2Seq model and other baselines, we consider two categories of evaluation criteria from the description generation quality and grounding accuracy respectively.

Description generation quality. We use 4 widely used metrics to evaluate the description generation quality. They are BLEU@4, METEOR, CIDEr and SPICE. These scores are calculated by the official evaluation scripts[1].

Grounding accuracy. Grounding accuracy is another metric to measure if a model can correctly predict both object words and their locations in video frames. It is measured by F1_all and F1_loc. The F1_all score measures the object words if they are correctly predicted and localized. And the F1_loc score only measures the correctly predicted object words. We also use the official evaluation scripts [2] [3] to measure all of these scores.

[1]https://github.com/ranjaykrishna/densevid_eval

[2]https://github.com/facebookresearch/ActivityNet-Entities

[3]https://competitions.codalab.org/competitions/20537

Method	B@4	C	M	S	F1_all	F1_loc
M. Trans	2.41	46.1	10.6	13.7	-	-
Temp-Attn	2.17	42.2	10.2	11.8	-	-
ZhouGVD	2.35	45.5	11.0	14.7	7.59	25.0
KNN-HAST	2.61	48.5	**11.3**	**15.1**	7.64	**26.5**
RG-HAST	**2.65**	**49.3**	11.2	**15.2**	**7.66**	26.1

Table 1: Results on Grounded ActivityNet-Entities test set. Notations: B@4-BLEU@4, C-CIDEr, M-METEOR, S-SPICE, M.Trans-Masked Transformer, TempAttn-BiLSTM+TempoAttn. All accuracies are in %.

p (KNN)	B@4	C	M	S	F1_all	F1_loc
5	2.70	49.4	11.2	15.2	7.04	23.5
10	**2.80**	**49.6**	**11.3**	**15.3**	**7.22**	**24.9**
20	2.76	49.4	**11.3**	15.2	6.91	23.4
30	2.71	49.3	11.2	14.9	6.89	23.5
40	2.68	48.9	11.1	15.1	6.70	23.2

Table 2: Results on Grounded ActivityNet-Entities val set.

4.4 Performance Comparisons

We compare HAST-Graph2Seq with the SOTA models, i.e., Masked Transformer [Zhou *et al.*, 2018], BiM-STM+TempoAtnn [Zhou *et al.*, 2018] and ZhouGVD [Zhou *et al.*, 2019] on Grounded ActivityNet Captions Dataset to verify the effectiveness of our method. Moreover, since the initial graph of the HAST-Graph2Seq can be constructed in two different ways, we also create two variants of HAST-Graph2Seq, i.e., KNN-HAST (the KNN initial graph with p is set to 10) and RG-HAST (the relational initial graph) to further investigate the relation between the different graph initializations and the final performance. For a fair comparison, we use the same C-3D video feature and the same region proposals extracted by Faster-RCNN pre-trained on Visual Genome (VG). For all these methods we performed the same experiments 3 times, and we reported their average scores.

As shown in Table 1, our methods outperform all the state-of-the-art models on all metrics, especially on CIDEr, BLEU@4, and SPICE, which highlights the importance of modeling relationships among the region proposal and using these relations for video description generation. Moreover, we can observe that the RG initialization outperforms the KNN initialization on most metrics. This suggests that an initial graph with external commonsense knowledge is beneficial to better modeling the relations among the regions and further improve the generation performance.

We further investigate the effect of KNN initialization with different number of neighbors by varying the KNN parameter p in $5, 10, 20, 30, 40$ on the validate set (the test set is not released so conducting on it is time-consuming). Table 2 shows how different KNN initialization effects on the final performance. From Table 2, we note that the HAST-Graph2Seq achieves the best performance when p lies around $5 - 20$. This suggests that for KNN initialization a proper p is crucial. When the p is too small, the initial graph may contain less useful relations, while when the p is too large, the graph

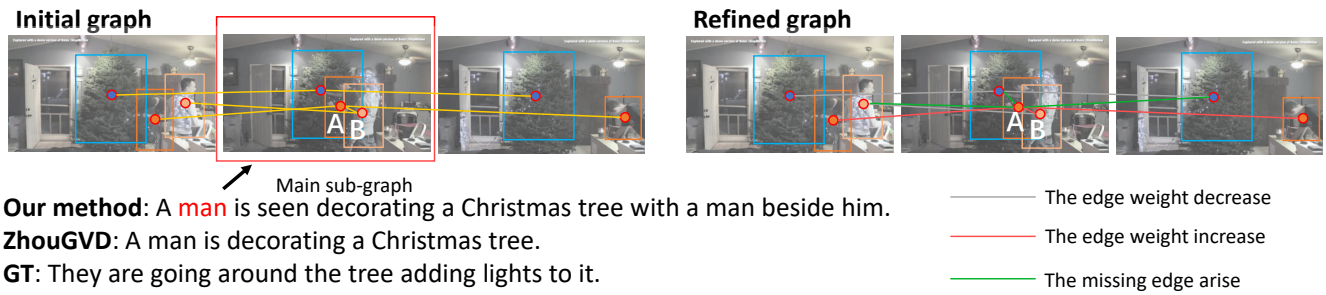

Figure 2: Qualitative differences between ZhouGVD and our proposed HAST with visualization of the discovered implicit graph structure.

Method	B@4	C	M	S	F1_all	F1_loc
ZhouGVD	2.59	47.5	11.2	15.1	7.11	24.1
KNN-HAST	**2.80**	**49.4**	**11.3**	**15.3**	**7.22**	**24.9**
-init.	2.60	47.4	11.0	14.7	6.65	22.4
-refine	2.70	48.1	11.1	14.8	6.83	23.5
-hie. attn.	2.70	48.8	11.2	15.0	6.91	23.7

Table 3: Results on Ablation Model on ActivityNet val set.

may contain too much noise to be refined.

4.5 Ablation Study

Next we conduct ablation studies to show how initial graph construction, graph refinement, and hierarchical attention contribute to the proposed method on the validate set. Without loss of generality, we consider KNN with $p = 10$ for initial graph construction. More concretely, we will discard one component at a time to generate ablation models as follows:

(1) w/o. initial graph (abbr: -init.). We remove the initial graph and use implicit graph generated by learned metrics.

(2) w/o. refinement (abbr: -refine). We remove the graph refinement component and use the KNN with $p = 10$ as the initial graph individually.

(3) w/o. hierarchical attention (abbr: -hie. attn.). We remove the hierarchical attention and replace it with the coarse-grain proposal attention proposed by [Zhou *et al.*, 2019].

Table 3 gives all ablation results on the validation set. As shown in Table 3, the HAST outperforms all ablation models on all metrics, which demonstrates that the initial graph construction, graph refinement, and hierarchical graph attention are all useful components for GVD.

Finally, by comparing among the ablation models, we find that a model without the initial graph performs the worst. This indicates that an good initial graph plays important role when exploiting spatial-temporal correlations in the video frames. The experiments also suggest that through refinement, the noise contains in the initial graph can be further reduced and the better graph topology can be discovered.

4.6 Qualitative Analysis

To qualitatively validate the effectiveness of our proposed HAST-Graph2Seq network, we present one typical example. Figure 2 shows the description results of our method, the best baseline ZhouGVD and the ground-truth on the Grounded ActivityNet Caption dataset, respectively. We can find that the baseline method misses the man B beside the man A, while our method can find the correct relation between them. The reason lies in two aspects. Firstly, when generating the first man, our model focused on the second sub-graph and then focused on the objects in it with the help of hierarchical attention. Thus, the model can find the semantic relation with the second man. Secondly, we visualize the initial KNN sequence graph and the refined sequence graph (we just show the second sub-graph as the key role and the main nodes and edges related to it for reasons of brevity). Through refinement, we can see that the weights of the key edges (eg. the man A with the man B) increase while the unimportant ones decrease. Thus, through our graph refinement techniques, we can discover optimal underlying graph structure that is important for video understanding.

5 Conclusion

In this paper, we propose a novel spatial-temporal sequence graph topology refinement with hierarchical attention for grounded video description task, which model the regions with spatial-temporal sequence graph. Specifically, we propose several methods to build the initial topology and refine it through end-to-end training. In addition, during decoding we apply hierarchical attention on the graph to focus on the regions in different gains. The extensive experiments demonstrate the effectiveness of our proposed method.

Acknowledgments

This work has been supported in part by National Key Research and Development Program of China (2018AAA010010), NSFC (U1611461, U19B2043, 61751209, 61976185), Zhejiang Natural Science Foundation (LR19F020002, LZ17F020001), University-Tongdun Technology Joint Laboratory of Artificial Intelligence, Zhejiang University iFLYTEK Joint Research Center, Chinese Knowledge Center of Engineering Science and Technology (CKCEST), Engineering Research Center of Digital Library, Ministry of Education, the Fundamental Research Funds for the Central Universities.

References

[Anderson *et al.*, 2018] Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. Bottom-up and top-down attention for image captioning and visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6077–6086, 2018.

[Chen *et al.*, 2019] Yu Chen, Lingfei Wu, and Mohammed J Zaki. Deep iterative and adaptive learning for graph neural networks. *arXiv preprint arXiv:1912.07832*, 2019.

[Chen *et al.*, 2020] Yu Chen, Lingfei Wu, and Mohammed J Zaki. Reinforcement learning based graph-to-sequence model for natural question generation. In *ICLR*, 2020.

[Gao *et al.*, 2019] Yuyang Gao, Lingfei Wu, Houman Homayoun, and Liang Zhao. Dyngraph2seq: Dynamic-graph-to-sequence interpretable learning for health stage prediction in online health forums. In *ICDM*, 2019.

[He *et al.*, 2016] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[Krishna *et al.*, 2017] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision*, 123(1):32–73, 2017.

[Li *et al.*, 2019a] Juncheng Li, Xin Wang, Siliang Tang, Haizhou Shi, Fei Wu, Yueting Zhuang, and William Yang Wang. Unsupervised reinforcement learning of transferable meta-skills for embodied navigation. *arXiv preprint arXiv:1911.07450*, 2019.

[Li *et al.*, 2019b] Linjie Li, Zhe Gan, Yu Cheng, and Jingjing Liu. Relation-aware graph attention network for visual question answering. *arXiv preprint arXiv:1903.12314*, 2019.

[Liu *et al.*, 2016] An-An Liu, Yu-Ting Su, Wei-Zhi Nie, and Mohan Kankanhalli. Hierarchical clustering multi-task learning for joint human action grouping and recognition. *IEEE transactions on pattern analysis and machine intelligence*, 39(1):102–114, 2016.

[Liu *et al.*, 2018] Anan Liu, Ning Xu, Hanwang Zhang, Weizhi Nie, Yuting Su, and Yongdong Zhang. Multi-level policy and reward reinforcement learning for image captioning. In *IJCAI*, pages 821–827, 2018.

[Ren *et al.*, 2015] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015.

[Tomei *et al.*, 2019] Matteo Tomei, Lorenzo Baraldi, Simone Calderara, Simone Bronzin, and Rita Cucchiara. Stage: Spatio-temporal attention on graph entities for video action detection. *arXiv preprint arXiv:1912.04316*, 2019.

[Tran *et al.*, 2015] 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.

[Vaswani *et al.*, 2017] 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.

[Venugopalan *et al.*, 2015] Subhashini Venugopalan, Marcus Rohrbach, Jeffrey Donahue, Raymond Mooney, Trevor Darrell, and Kate Saenko. Sequence to sequence-video to text. In *Proceedings of the IEEE international conference on computer vision*, pages 4534–4542, 2015.

[Wang and Gupta, 2018] Xiaolong Wang and Abhinav Gupta. Videos as space-time region graphs. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 399–417, 2018.

[Xu *et al.*, 2018a] Kun Xu, Lingfei Wu, Zhiguo Wang, Yansong Feng, Michael Witbrock, and Vadim Sheinin. Graph2seq: Graph to sequence learning with attention-based neural networks. *arXiv preprint arXiv:1804.00823*, 2018.

[Xu *et al.*, 2018b] Ning Xu, An-An Liu, Yongkang Wong, Yongdong Zhang, Weizhi Nie, Yuting Su, and Mohan Kankanhalli. Dual-stream recurrent neural network for video captioning. *IEEE Transactions on Circuits and Systems for Video Technology*, 29(8):2482–2493, 2018.

[Yang *et al.*, 2019] Xu Yang, Kaihua Tang, Hanwang Zhang, and Jianfei Cai. Auto-encoding scene graphs for image captioning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 10685–10694, 2019.

[Yao *et al.*, 2018] Ting Yao, Yingwei Pan, Yehao Li, and Tao Mei. Exploring visual relationship for image captioning. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 684–699, 2018.

[Zhang and Peng, 2019] Junchao Zhang and Yuxin Peng. Object-aware aggregation with bidirectional temporal graph for video captioning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8327–8336, 2019.

[Zhou *et al.*, 2018] Luowei Zhou, Yingbo Zhou, Jason J Corso, Richard Socher, and Caiming Xiong. End-to-end dense video captioning with masked transformer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8739–8748, 2018.

[Zhou *et al.*, 2019] Luowei Zhou, Yannis Kalantidis, Xinlei Chen, Jason J Corso, and Marcus Rohrbach. Grounded video description. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6578–6587, 2019.