Learning Coarse-to-Fine Graph Neural Networks for Video-Text Retrieval

Wei Wang, Junyu Gao, Xiaoshan Yang, and Changsheng Xu, *Fellow, IEEE*

***Abstract*—We address the problem of video-text retrieval that searches videos via natural language description or vice versa. Most state-of-the-art methods only consider cross-modal learning for two or three data points in isolation, ignoring to get benefit from the structural information of other data points from a global view. In this paper, we propose to exploit the comprehensive relationships among cross-modal samples via Graph Neural Networks (GNN). To improve the discriminative ability for accurately finding the positive sample, a Coarse-to-Fine GNN is constructed, which can progressively optimize the retrieval results via multi-step reasoning. Specifically, we first adopt heuristic edge features to represent relationships. Then we design a scoring module in each layer to rank the edges connected to the query node and drop the edges with lower scores. Finally, to alleviate the class imbalance issue, we propose a random-drop focal loss to optimize the whole framework. Extensive experimental results show that our method consistently outperforms the state-of-the-arts on four benchmarks.**

***Index Terms*—video-text retrieval, graph neural network, coarse-to-fine strategy.**

I. Introduction

WITH the widely use of mobile phones and portable cameras, tons of user-generated videos are uploaded every day to video sharing websites, such as YouTube, Flickr and Vimeo. A powerful query-based ad-hoc video search method is essential to find the favorite videos for front-end users while the capability of finding a proper natural language description of the video is also demanded in video categorization and recommendation tasks for back-end servers. Due to the potential applications, video-text retrieval [1], which means retrieving videos via natural language descriptions or vice versa, has attracted remarkable attention in recent years. Nevertheless, the semantic gap between the natural language sentence and the video is still a crucial bottleneck. A video has not only various scenes, actions and moving objects, but also complex spatial and temporal structures [2, 3]. Though the language sentence is much simpler to process than the video data, capturing the high-level semantic structures of the cross-modal data is still a difficult problem in matching the textual and video contents.

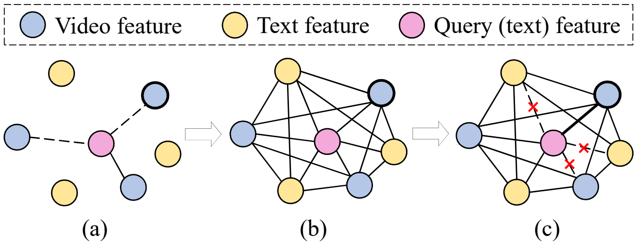


Fig. 1: *Three types of strategies for video-text retrieval.* Take text-to-video retrieval as an example. (a) Conventional strategy only considers two or three samples in similarity learning. (b) GNN-based strategy exploits structural relationships among all the samples. (c) Our CF-GNN not only models the structural information but also progressively improves its discriminative ability by focusing on the positive sample in a coarse-to-fine manner.

Video-text retrieval has been extensively investigated in recent years with the success of deep learning in computer vision and natural language processing. Most of the these methods [4–6] aim to learn compact feature representations of videos and texts, then jointly embed them into a common space where the relevant video-text pairs are close and irrelevant pairs are far away. For example, the state-of-the-art video retrieval method, Dual Encoding [7], uses multi-level encodings of videos and texts and trains the model in a common space using triplet loss. Another SOTA method Collaborative Experts [8] adopts various features on pretrained experts like audio track and OCR and embeds them into the joint space via a collaborative gated module where a bidirectional max-margin ranking loss is deployed.

Although these approaches achieve promising results, as shown in Figure 1 (a), they only consider the cross-modal relationships among two or three data points in each learning step and use contrastive or triplet loss as objective, while ignoring to exploit the structural information of other data points from a global view. In fact, diverse data points can provide rich context information for cross-modal similarity learning and preserve their intrinsic semantic structures. However, incorporating semantic relations and structural similarities of the whole data space into cross-modal retrieval is not trivial. Despite deep metric learning-based methods [9, 10] utilize

Manuscript received *** **, 2020; revised *** **, 2020; accepted *** **, 2020. This work was supported by National Key Research and Development Program of China (No. 2018AAA0102200), National Natural Science Foundation of China (No. 61720106006, 61721004, 61832002, 61702511, 61751211, 61532009, U1836220, U1705262, 61872424, 61936005) and Key Research Program of Frontier Sciences of CAS (QYZDJSSWJSC039). This work was also supported by Research Program of National Laboratory of Pattern Recognition (No. Z-2018007).

Wei Wang, Junyu Gao, Xiaoshan Yang, and Changsheng Xu are with the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China, and with the School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing 100049, China, and also with the PengCheng Laboratory, Shenzhen 518066, China. (e-mail: wangwei2018@ia.ac.cn; gaojunyu2015@ia.ac.cn; xiaoshan.yang@nlpr.ia.ac.cn; csxu@nlpr.ia.ac.cn).

several losses to constrain the query data to be closer to the positive data than the negative ones, they cannot explicitly model the relationships between any two data points for structural learning. Therefore, our first question is: *How to explicitly leverage the relationships between different data points for structural video-text retrieval?*

Recently, Graph Neural Networks(GNNs), which receive significant attention in many computer vision fields [11–15], are able to handle data with rich relational structures by aggregating and updating neighbouring features iteratively. By learning the dependencies and propagating messages between any two nodes in an arbitrary graph, GNNs have great potential for video-text retrieval. Intuitively, as shown in Figure 1 (b), taking a query as well as a set of videos and texts as graph nodes, we can construct a cross-modal graph and directly employ the popular GNNs like Graph Convolutional Networks (GCNs) [16] to model the structural information. However, in such a graph, a large amount of relationships should be considered, which brings difficulty in finding the matched target data corresponding to the query. In fact, for a given query, there may only exist one positive sample while others are negative. Such negative samples may distract the model from focusing on the most relevant one. As a result, our second question is: *How to design an effective graph neural network that can not only model the structural information but also improve the discriminative ability by focusing on the positive data?*

To address the above two questions, as shown in Figure 1 (c), we propose a Coarse-to-Fine Graph Neural Network (CF-GNN) for video-text retrieval, which can progressively improve the model's discriminative ability and locate the positive sample via multi-step reasoning. Specifically, as shown in Figure 2, for a given query (either a text or video), we build a cross-modal graph where the nodes are videos and texts. To reduce the computational cost and avoid the influence of irrelevant data, we adopt a pretrained cross-modal feature extractor to retrieve the top-K relevant data samples as graph nodes. For designing edges that connect nodes, we adopt multi-dimensional edge features to depict the complex and underlying relationships among videos and texts. Note that some previous works [17, 18] have shown that exploiting edge features achieves better performance than others with scalar-based adjacency matrix [16, 19]. Here, we initialize the edge features with heuristic information (*e.g.*, cosine distance, Manhattan distance and Euclidean distance) calculated from the connected nodes,which can capture rich relationships and improve the discriminative ability of similarity learning. To perform coarse-to-fine refinement, in each layer of our framework, we design a scoring module to rank the edge features connected to the query node and drop the edges with lower scores. With the multi-step reasoning process in our GNN, the discriminative edges are preserved, which makes the retrieval results better in the deeper layers. To optimize the whole framework, we propose a random-drop focal loss in each layer to balance positive and negative samples and mine hard samples. Extensive experimental results demonstrate the significant performance of the proposed CF-GNN.

Our main contributions can be summarized as follows:

- We propose a novel coarse-to-fine graph neural network for video-text retrieval which employs GNNs to model the structural similarities among videos and texts and progressively find the positive sample. To the best of our knowledge, our CF-GNN is among the first to use GNNs for video-text retrieval with a coarse-to-fine strategy.
- By carefully designing the heuristic edge features, learnable scoring modules, and random-drop focal loss, our proposed method can jointly enjoy the merits of highly discriminative ability, robust relationship refinement, and balanced training of positive and negative samples.
- We have verified the effectiveness of CF-GNN on three popular video-text retrieval benchmarks. Our approach consistently outperforms the state-of-the-art. We also evaluate the performance on image-text retrieval tasks. The favorable results on COCO dataset [20] demonstrate the promising potential of our framework.

II. Related Work

A. Video-Text Retrieval

Most video-text retrieval frameworks aim to learn video and text features in a common space where similarities can be calculated. For video feature representation, early approaches adopt handcrafted features like HSV Color Histograms [21–23], Local Binary Patterns (LBP)[22, 24], SIFT [25, 26], and so on. In recent years, with powerful deep learning techniques, most newly proposed video-text retrieval approaches employ varieties of CNN and RNN architectures for better performance. Yu *et al.* [27] use Long Short Term Memory (LSTM) modules to encode sequence of video frames. Dual Encoding [7] employs a bi-GRU and a pretrained CNN to encode video features at global, temporal and local levels. For text features, [4] formulates sentences as subject-verb-object triplets by RNN. W2VV [28] proposes a multi-scale sentence vectorization that utilizes Bag-of-Words (BoW), word2vec and RNN based text encodings. W2VV++ [29] improves W2VV with a better designed sentence encoding strategy which considers the outputs from all intermediate steps of GRUs. Variations of RNNs are also explored in text feature extracting in [1, 30]. Other methods, like [31], get additional useful topic facets from the Internet to improve the retrieval performance.

For the similarity learning, [32], one of the earliest works, averages the frame-level features of one video and computes a dot product with others to measure the similarity of whether they belong to the same query text. BoW approach [33] maps each frame-level feature into one or more visual words and generates a tf-idf representation, based on which the video-text retrieval is performed by calculating cosine distance. Hashing is another research line which encodes videos into the Hamming codes via learned hash functions [21, 34–36]. Recently, much progress in joint video-language embedding is made by extending successful image-text retrieval methods like VSE++ [1] or leveraging stronger deep models like bi-GRU to extract global, local and temporal patterns [1, 27, 30]. For instance, [5] proposes an LSTM with visual-semantic embedding method that jointly minimizes the distance between video and text vectors in the common space. However, structural similarities among all data samples in common space are not comprehensively considered in video-text retrieval before, which leads to a decrease in retrieval performance. The coarse-to-fine retrieval strategy also remains to be explored.

B. Graph Neural Networks

There are many kinds of non-Euclidean data structures in the real world such as social network and biogenic protein structure, which cannot be modeled by conventional deep architectures like CNNs or RNNs. Graph Neural Network [37] is a powerful deep architecture that captures complex interactions among non-Euclidean data and mines latent information. Currently, many GNNs focus on learning node features but ignore learning multi-dimensional edge features. For instance, Garcia and Bruna [38] propose a densely connected graph in few-shot learning task, where every input node of the graph represents the embedding feature and label information. Gao *et al.* [39] employ a GNN in video classification task where nodes in the graph are concepts of videos. Xu *et al.* [34] propose a Graph Convolutional Hashing (GCH) approach, which learns modality-unified binary codes via an affinity graph. Recently, the potential of using edge features is explored. [17] proposes a novel model that is capable of exploiting multi-dimensional edge features, which obtains better performance compared with conventional Graph Convolutional Graphs (GCNs) [16]. Kipf *et al.* [40] propose to infer the interactions and learn the dynamics from the observational data with interpretable edge features. Kim *et al.* [18] propose edge-labeling GNN (EGNN) which explicitly exploits edge features and significantly improves the performance over the existing GNNs. Our proposed model also focuses on edge features. Unlike EGNN that only utilizes 2-D edge features, we employ high-dimensional edge features to hold rich relationships among data points.

Recently, there are several methods applying GNNs for cross-modal retrieval. Chen *et al.* [41] propose a Hierarchical Graph Reasoning (HGR) model for video-text retrieval. The HGR generates hierarchical textual embeddings with modified relational GCNs and aligns texts with videos at different levels, while our model utilizes the global textual and visual embeddings for alignment. However, the HGR model only exploits the structural similarities on the text side and ignores the complex relationship among texts and videos. Besides, in terms of inference, the massive local pair matching at different levels in HGR is time-consuming. [42] is another method that uses GCNs to perform text feature extraction only. GCH [34] employs a GCN to excavate data structural information to generate optimal hash codes. However, each node in GCN is a feature vector which is fused from a cross-modal pair. Therefore, GCH cannot fully model the relations among all data points. Besides, GCH does not fully exploit edge features and lacks a coarse-to-fine retrieval strategy. It is also worth mentioning that there are some *k*-NN graph-based methods for cross-modal retrieval. [43, 44] build a graph utilizing the top-*k* most relevant image features to the query image and update the graph features based on hand-crafted rules. These approaches do not consider multi-modal information and the graph is not learned in an end-to-end manner. Other methods like [45] construct the *k*-NN graph whose nodes are cross-modal data points while the graph has no parameters to learn and just functions to guide the training of another module.

III. Method

Video-text retrieval consists of text-to-video (text as queries) and video-to-text (video as queries) retrieval, which are symmetric in our model. For simplicity, we will explain our framework based on text-to-video retrieval, which we call video retrieval in short. As illustrated in Figure 2, CF-GNN is a coarse-to-fine video-text retrieval framework that progressively improves the mode's discriminative ability and locates the positive sample via multi-step reasoning. Details are described as follows.

A. Graph Construction

Given a video feature set $\{\mathbf{v}_i\}$ and a text feature set $\{\mathbf{t}_i\}$ produced by a pretrained cross-modal feature extractor in *d*-dimensional common space, a query feature $\mathbf{q} \in \{\mathbf{t}_i\}$ is randomly selected from the text feature set (or from the video feature set in video-to-text retrieval). In practical scenarios, the number of videos and texts can be extremely large, which makes it impossible to take all of them into account at one time. To reduce the computational cost and avoid the influence of irrelevant data, we retrieve the top-K videos and top-K texts to the query based on cosine similarities.

To model the structural information among these data points, we design a fully-connected graph whose nodes are the retrieved $2K+1$ videos, texts, and query. The edges of the graph can be learned by our proposed CF-GNN. Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be the constructed graph, where $\mathcal{V}$ and $\mathcal{G}$ denote the node set and edge set, respectively. The node features are initialized with the *d*-dimensional features of the videos, texts, and query:

$$\{\mathbf{x}_i^0\}_{i=1\ldots 2K+1} = \{\mathbf{v}_i\}_{i=1\ldots K} \cup \{\mathbf{t}_i\}_{i=1\ldots K} \cup \{\mathbf{q}\} \qquad (1)$$

To capture complex relationships among nodes, we aim to learn multi-dimensional edge features. In our framework, the edge features are initialized with heuristic information that measures the similarities between the connected nodes in different manners. Specifically, the edge connected to node $\mathbf{x}_i$ and $\mathbf{x}_j$ is initialized as:

$$\mathbf{e}_{ij}^0 = [\text{cosine}(\mathbf{x}_i^0, \mathbf{x}_j^0) || \ell_1(\mathbf{x}_i^0, \mathbf{x}_j^0) || \ell_2(\mathbf{x}_i^0, \mathbf{x}_j^0)] \qquad (2)$$

where $||$ is the concatenation operation. The cosine, ℓ_1, ℓ_2 represent cosine, Manhattan, and Euclidean distance, respectively. The combination of these metrics holds a more comprehensive relationship between the nodes. The dimension of edge features will increase in message passing, enabling edge features to hold richer relationships.

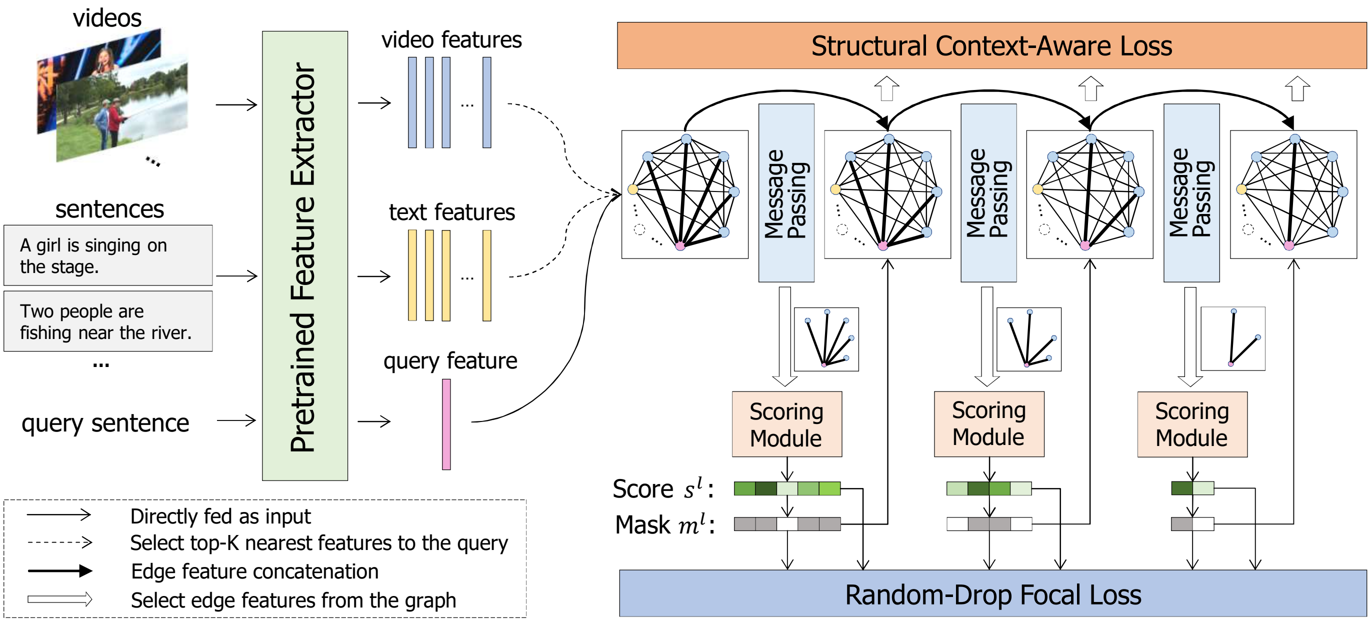

videos
sentences
A girl is singing on the stage.
Two people are fishing near the river.
...
query sentence
Pretrained Feature Extractor
video features
text features
query feature
Structural Context-Aware Loss
Message Passing
Message Passing
Message Passing
Scoring Module
Scoring Module
Scoring Module
Score s^l:
Mask m^l:
Random-Drop Focal Loss
Directly fed as input
Select top-K nearest features to the query
Edge feature concatenation
Select edge features from the graph

Fig. 2: *Overview of the proposed CF-GNN (text-to-video retrieval).* The input are the videos, corresponding sentences and the query sentence. We first use a pretrained feature extractor to extract features in d-dimensional common space. Then we construct the graph by selecting the query feature, Top-K nearest video features and text features respectively as node features. The graph is fully-connected where edge features are initialized with heuristic information of the connected nodes. We apply message passing and learn a scoring module on selected edge features for coarse-to-fine refinement in each layer. Finally, we employ the random-drop focal loss and structural context-aware loss to optimize the model.

B. *Our Proposed CF-GNN*

Since we have constructed the graph with node features and edge features, the problem now lies in how to pass messages between nodes and conduct coarse-to-fine retrieval.

Message Passing. The CF-GNN consists of L layers where the node features and edge features are updated alternately in each layer. To be specific, in layer $l+1$, we first update each node feature $\mathbf{x}_i^{l+1}$ by aggregating all its neighbour nodes along with the connected edge features. Since each element in an edge feature can be regarded as a similarity measurement between connected nodes, we can update node features as follows:

$$\mathbf{x}_i^{l+1} = f_x^{l+1}(\mathbf{x}_i^l || \frac{1}{D^l}(\sum_{d=1}^{D^l}\sum_{j\neq i} \frac{\mathbf{e}_{ijd}^l}{\sum_k \mathbf{e}_{ikd}^l}\mathbf{x}_j^l)) \tag{3}$$

where $\mathbf{e}_{ijd}$ is the d_{th} element of the edge feature vector $\mathbf{e}_{ij}$, D^l is the length of edge feature vector in the l_{th} layer, and f_x^{l+1} is the node feature update network. By synthetically measuring the weighted proportions of similarities between all neighbouring nodes, we update the node features via considering the structural information.

Then edge features are updated based on the newly updated node features. We employ an edge feature update network f_e to measure the similarities between node $\mathbf{x}_i$ and all its neighbour nodes. Specifically, for the element $\mathbf{e}_{ijd}$, we can update it according to the weighted proportions of all the connected edge features as follows:

$$\bar{\mathbf{e}}_{ijd}^{l+1} = \frac{\mathbf{h}_{ijd}^{l+1}\mathbf{e}_{ijd}^l}{\sum_k \mathbf{h}_{ijd}^{l+1}\mathbf{e}_{ikd}^l}\sum_k \mathbf{e}_{ikd}^l \tag{4}$$

$$\mathbf{e}_{ij}^{l+1} = \mathbf{e}_{ij}^l || \bar{\mathbf{e}}_{ij}^{l+1} \tag{5}$$

where $\mathbf{h}_{ij}^{l+1} = f_e^{l+1}(|\mathbf{x}_i^{l+1} - \mathbf{x}_j^{l+1}|)$, f_e^{l+1} is the edge feature update network whose output has the same dimension as $\mathbf{e}_{ij}^l$. Here, the learned $\mathbf{h}_{ij}^{l+1}$ provides additional attention on the edge features in the l_{th} layer. A similar update rule can be found in [18]. Asymmetric edges can better model the complex structural relationships. Then we concatenate $\bar{\mathbf{e}}_{ij}^{l+1}$ with the edge features in the previous layer, which doubles the dimension of edge features. The concatenation brings several benefits which will be explained later. Note that although we can adopt different edge update networks to measure different types of relationships such as text-text, text-video and video-video, we find this design does not obtain much gain in our experiments. For simplicity, we utilize the same f_e^{l+1} for all the edge types.

Coarse-to-Fine Edge Refinement. We build a coarse-to-fine video-text retrieval framework that progressively improves the model's discriminative ability and locates the positive sample via multi-step reasoning. Based on the retrieval results in the previous layer, CF-GNN produces retrieval results which are more refined in the current layer. To be specific, for the l_{th} layer, we aim to obtain a binary refining mask to record the selection of edges connected to the query node and video nodes. The edge selection can also be viewed as a selection of candidate videos. With the edge features $\mathbf{e}^l$, we design a scoring module S^l to rank the candidate videos:

$$\mathbf{s}^l_{qv_i} = S^l(\mathbf{e}^l_{qv_i}) \tag{6}$$

where q and v_i denote the index of the query and the i_{th} candidate video, respectively. $\mathbf{s}^l_{qv_i}$ is the importance score for retaining the corresponding edge. Since the edge features in the current layer are concatenated with the edge features from the previous layer, our method can avoid a drastic variation between the learned scores in consecutive layers.

Intuitively, the edges with higher scores are more likely to be connected to the target video. We consider the top-k^l candidate videos for further coarse-to-fine processing. As a result, we can generate a refining mask $\mathbf{m}^l$ as follows:

$$\mathbf{m}^l_{qv_i} = \begin{cases} 1, & \text{rank}(\mathbf{s}^l_{qv_i}) \le k^l \\ 0, & \text{else} \end{cases} \tag{7}$$

where $\mathbf{m}^l_{qv_i}$ is the i_{th} element of the refining mask for the query q. The rank$(\cdot)$ is a function which indicates the sorted index of the input. $\mathbf{m}^l$ records a newly refined candidate video selection produced in this layer and will be sent to the next layer as a coarse selection to be further refined.

We repeat the above steps in each layer, generating a set of coarse-to-fine retrieval results layer by layer and progressively improve the discriminative ability of our model.

C. Learning and Inference

Loss Function. Training the coarse-to-fine video-text retrieval framework described above is not trivial. In fact, the edge refinement in each layer should be valid thus can ensure the effectiveness of the coarse-to-fine strategy. To this end, we add training objectives into each layer and adopt a warmup learning strategy (see section IV-B). This design also avoids the gradient vanishing problem in the deep architecture. Besides, residual connections that concatenate edge features over layers, also accelerates the gradient flow. In the training process, each layer in our framework, which can be viewed as a coarse-to-fine step, is trained to select a more refined candidate video set where fewer negative samples are included and the positive sample is located more precisely. Due to the severe class imbalance of positive samples and negative samples, for an edge score s^l_{ij}, we define a random-drop focal loss in each layer as follows:

$$\begin{aligned}\hat{L}^l_{ij} = &-\mathbf{m}^l_{ij}[\alpha^l Y_{ij}(1-\mathbf{s}^l_{ij})^\gamma \text{log}\mathbf{s}^l_{ij} \\ &+ \mathbf{w}^l_{ij}(1-\alpha^l)(1-Y_{ij})(\mathbf{s}^l_{ij})^\gamma \text{log}(1-\mathbf{s}^l_{ij})]\end{aligned} \tag{8}$$

where Y_{ij} is the ground-truth label of the edge $\mathbf{e}_{ij}$, $Y_{ij} = 1$ means the data points i and j are matched (see Section IV-B for more details), α^l is a weight balancing positive samples and negative samples, $\mathbf{w}^l_{ij}$ is a binary code drawn from a Bernoulli distribution which takes the value 1 with probability p^l. $\mathbf{m}_{ij}$ is the mask meaning that we focus on the refined samples. With Eq. (8), the loss of all the query-video pairs in layer l is:

$$\mathcal{L}^l_{qV^l} = \sum_{v_i \in V^l} \hat{L}^l_{qv_i} \tag{9}$$

where q and V^l are the query and candidate video set in layer l, respectively. $\hat{L}_{qv_i}$ is calculated following Eq. (8). By randomly dropping some negative samples in the selected candidate video set, our loss function focuses on the most relevant positive samples and part of the negative candidate video samples for better training. The random-drop operation in the loss can further assist the refinement in coarse-to-fine steps: CF-GNN excludes some candidate videos that are least likely to be the target sample in each layer using the learned mask $\mathbf{m}$, while the random-drop operation in our loss function further drops some randomly selected negative samples to help alleviate class imbalance and the impact of noise samples.

Up to now, we have only considered the query-video relationships, while ignoring other context information such as video-text, text-text relationships (note that there is no video-video pair as videos are independent). In fact, the contextual videos and texts around the query share structural semantics, which provide useful information for locating the target video. Besides, there are more matched pairs in video-text and text-text relationships, which helps alleviate the class imbalance problem. In our experiment, we find that further exploiting the relationships among these data points improves the retrieval performance (see section IV-D). To avoid computational cost, we select the top-c most similar videos V_c and texts T_c to the query and constrain all the additional video-text, text-text relationships using Eq. (8) and calculate the structural context-aware loss:

$$\begin{aligned}\mathcal{L}^l_{\text{context}} &= \mathcal{L}^l_{V_c T_c} + \mathcal{L}^l_{T_c T_c} \\ &= \sum_{v_i \in V_c} \sum_{t_j \in T_c} \hat{L}^l_{v_i t_j} + \sum_{t_i \in T_c} \sum_{t_j \in T_c, j \neq i} \hat{L}^l_{t_i t_j}\end{aligned} \tag{10}$$

We find that learning a refining mask $\mathbf{m}$ for structural context-aware loss does not obtain much gain in our experiments, thus we set $\mathbf{m}$ as 1. Overall, the total loss for each constructed graph in our framework can be formulated as:

$$\mathcal{L} = \sum_l [\lambda \mathcal{L}^l_{qV^l} + (1-\lambda)\mathcal{L}^l_{\text{context}}] \tag{11}$$

where λ is the trade-off hyperparameter.

Inference. At test phase, given a query, we use the masked scores in the last layer of GNN to rank the candidate samples. Since CF-GNN has been well trained, it is capable of capturing structural similarities among data points and refining the retrieval results layer by layer. The candidate sample with the highest score in the last layer of GNN is the final retrieval result.

D. Discussion

Is it a general retrieval model and not specified to video-text retrieval? As the first step of using GNNs for video-text retrieval, we put more concentration on the structural modeling than video/text representation learning. Compared with other retrieval tasks like image-text retrieval, video-text retrieval is more challenging and has a larger domain gap. Also, the structural relationships among videos and texts are extremely complex, which requires a well-designed model as a remedy. Therefore, our work mainly aims to narrow the gap on this challenging task. Note that designing specific feature representations for different modalities is not our major goal, while we mainly focus on the complex relationship modeling and refinement. Obviously, the proposed framework has a good generalization ability for other retrieval tasks.

IV. Evaluation

We conduct extensive experiments on 4 popular benchmarks to evaluate our framework. For video-text retrieval, we first evaluate our CF-GNN on the MSR-VTT dataset [46] following the settings in [7]. Then we replicate the experiments on MSVD dataset [47] and TGIF dataset [48] to evaluate cross-dataset generalization of our model. To test the robustness of our method, we adopt different pretrained feature extractors [7, 8] for evaluation. For other cross-modal tasks, we perform text-to-image and image-to-text retrieval on MS-COCO dataset [20] with the feature extractor used in VSE++ [49] to evaluate cross-domain generalization. The favorable results demonstrate the high generalization performance of our method.

A. Datasets and Evaluation Metric

Four benchmark datasets are included in our experiments, among which MSR-VTT, MSVD, TGIF are video datasets with corresponding descriptions and MS-COCO is an image dataset consisting of corresponding captions.

MSR-VTT. The MSR-VTT dataset is a large-scale video benchmark for video understanding, providing 10k video clips and 200k corresponding clip-sentence pairs. Each clip is annotated with 20 descriptions in average. According to the official data partition, 6513 clips are used for training, 497 clips for validation and 2990 clips for testing. MSR-VTT is one of the largest datasets in video-text retrieval.

MSVD. The MSVD dataset contains 1970 clips from YouTube and each video has about 40 desriptions in multiple languages. We use English sentences only and the same data splits as previous work [50] for a fair comparison. Specifically, we use 1200 clips for training, 100 clips for validation and the remaining 670 clips for testing.

TGIF. The TGIF dataset contains 100k animated GIFs and 120k description sentences. The animated GIFs can be seen as video clips and are suggested using video description techniques. There is 1 sentence per animated GIF for the training and validation splits, and 3 sentences per GIF for the test split. According to the official data splits, we use 80,000 GIFs for training, 10,708 GIFs for validation and 11360 GIFs for testing.

MS-COCO. MS-COCO is a large-scale object detection, segmentation, and captioning dataset. Each image has 5 captions. In image-text retrieval, we use the split of [7], where the training set contains 82,783 images, 5000 images for validation and 5000 images for testing.

Evaluation Metric. We follow the standard evaluation criterion used in most prior video-text retrieval works [1, 7, 8]. We measure the rank-based performance, namely R@K (K=1,5,10), Median rank (Medr), Mean rank (Meanr) and mean Average Precision (mAP). R@K is the proportion that at least one correct item to the query is found among the top-K retrieved results. Median rank and Mean rank are the median position and mean position of the first correct item of the results. Higher R@K, mAP and lower Medr, Meanr are expected in video-text retrieval.

B. Implementation

We implement our framework using PyTorch [51] and run on TITAN RTX GPUs. Our CF-GNN consists of 3 layers since we found adding more layers can not bring much gain but the extra computational and space overhead will slow down the inference speed. The size of video features and text features generated from the pretrained cross-modal feature extractor is 2048 (*d*=2048). In graph construction, we select the top-100 videos and texts ($K = 100$) which are most similar to the query in all datasets, which can cover the target sample for most queries. We follow the same structures of node feature update network f_x and edge feature update network f_e in [18], except the dimensions of output. The dimensions of edge features in each layer, D^l, are 6, 12, 24, respectively while the initial edge features are 3-dimensional vectors. The node dimension is compressed from 2048 to 256 in the first layer and remains unchanged. The scoring module S is a fully-connected layer with batch normalization and sigmoid activation function. Experiments show that our model is robust to the parameter k^l in each layer and it will be analyzed later (see section IV-D). We set k^l to 90, 60, 30 respectively and our model works well. The ground-truth label Y_{ij} in loss function is defined as follows: For i, j in the same modality, it is set to 1 for texts only when both of them are the descriptions of the same video and set to 0 for all videos since videos are independent in the datasets; for i, j in different modalities, it is set to 1 only when they are a matched video-text pair. The parameter c related to $\mathcal{L}_{\text{context}}$ is set to 5. The α, γ, p and λ in loss function are set to 0.8, 2, 0.7 and 0.7 empirically in each layer. For the video-text retrieval task, we employ the pretrained video and text feature extractors used in [7, 8] to obtain the initial video and text representations. For the image-text retrieval, we employ the pretrained image and text feature extractor in [49]. We use SGD with Adam [52] optimizer and set initial learning rate to 0.0001. For every 5 epochs we decrease the learning rate by half. The weight decay is 10^{-5}. The mini-batch size is set to 13 in order to make full use of the GPU memory. In the training process, we do not use the refining masks in the first two epochs in order to warm up our CF-GNN model. If not, the initial model cannot figure out which candidate videos to be preserved during the coarse-to-fine step at the very beginning.

TABLE I: **State-of-the-art on MSR-VTT.** Larger R@1,5,10, mAP and smaller Med r, Mean r indicate better performance.

Method	Text-to-Video Retrieval						Video-to-Text Retrieval					
	R@1	R@5	R@10	Med r	Mean r	mAP	R@1	R@5	R@10	Med r	Mean r	mAP
VSE[49]	5.0	16.4	24.6	47	215.1	-	7.7	20.3	31.2	28	185.8	-
VSE++[1]	5.7	17.1	24.8	65	300.8	-	10.2	25.4	35.1	25	228.1	-
Mithun et al.[1]	5.8	17.6	25.2	61	296.6	-	10.5	26.7	35.9	25	266.6	-
W2VV[28]	6.1	18.7	27.5	45	-	0.131	11.8	28.9	39.1	21	-	0.058
Dual Encoding[7]	7.7	22.0	31.8	32	206.6	0.155	12.8	30.4	42.3	16	123.2	0.065
Ours + Dual Encoding	**8.0**	**23.2**	**32.6**	**31**	**206.0**	**0.160**	**14.3**	**32.2**	**44.3**	**14**	**121.3**	**0.069**
CE[8]	22.5	52.1	65.5	5	22.5	0.365	34.4	64.6	77.0	3	13.2	0.213
Ours + CE	**24.3**	**56.6**	**70.4**	**4**	**20.4**	**0.392**	**37.9**	**68.1**	**79.1**	**2**	**10.9**	**0.251**

TABLE II: **State-of-the-art on MSVD.** Larger R@1,5,10 and smaller Med r, Mean r indicate better performance.

Method	Text-to-Video Retrieval					Video-to-Text Retrieval				
	R@1	R@5	R@10	Med r	Mean r	R@1	R@5	R@10	Med r	Mean r
CCA[53]	-	-	-	-	251.3	-	-	-	-	245.3
JMDV[4]	-	-	-	-	224.1	-	-	-	-	236.3
VSE[49]	12.3	30.1	42.3	14	57.7	15.8	30.2	41.4	12	84.8
VSE++[49]	15.4	39.6	53.0	9	43.8	21.2	43.4	52.2	9	79.2
Dual Encoding[7]	21.6	49.5	62.3	**6**	34.7	27.8	48.7	58.7	6	55.5
Ours + Dual Encoding	**22.8**	**50.9**	**63.6**	**6**	**34.2**	**30.9**	**54.2**	**63.7**	**4**	**42.2**

TABLE III: **State-of-the-art on TGIF.** Larger R@1,5,10 and smaller Med r, Mean r indicate better performance.

Method	Text-to-Video Retrieval					Video-to-Text Retrieval				
	R@1	R@5	R@10	Med r	Mean r	R@1	R@5	R@10	Med r	Mean r
VSE++[49, 54]	1.55	5.89	9.77	220	-	1.42	5.63	9.60	192	-
Order[54, 55]	1.58	5.57	9.41	205	-	1.67	5.49	9.20	223	-
Corr-AE[54, 56]	2.10	7.38	11.86	148	-	2.15	7.29	11.47	158	-
DeViSE[6, 54]	1.58	5.57	9.41	205	-	1.67	5.49	9.20	223	-
PVSE[54]	3.01	9.70	14.85	109	-	3.28	9.87	15.56	115	-
Dual Encoding[7]	9.5	22.3	30.1	46	382.7	12.9	28.2	37.2	25	239.7
Ours + Dual Encoding	**10.2**	**23.0**	**30.7**	**44**	**382.2**	**13.4**	**29.4**	**38.5**	**23**	**239.0**

C. Results

Video-Text Retrieval. On MSR-VTT benchmark, we compare our proposed method with existing state-of-the-art methods in text-to-video retrieval and video-to-text retrieval tasks. We adopt two baseline feature extractors, the first is Dual Encoding [7], which only exploits RGB information to extract the video features. The second is Collaborative Experts (CE) [8] that employs multi-modal information such as audio, OCR, and face recognition for representing a video.

Table I summarizes the performance of CF-GNN on MSR-VTT dataset. We observe that our proposed method consistently outperforms other state-of-the-art video-text retrieval methods in all evaluation metrics. Specifically, our method obtains the mAP of 0.392 and 0.251 in text-to-video and video-to-text retrieval respectively, which significantly outperforms the 0.365 and 0.213 by CE. The results suggest that CF-GNN can fully exploit structural similarities among query, videos and texts and generate more accurate retrieval results. Besides, CF-GNN performs well on different pretrained cross-modal feature extractors using different kinds of video features, showing a good generalization of our model. Finally, we observe that compared with the state-of-the-art methods using RGB information of videos, CF-GNN obtains more improvements on those using multi-modal information of videos. This is because the multi-modal information of videos contains richer discriminative information which can be further exploited by CF-GNN.

On MSVD dataset, we retrain the Dual Encoding model on MSVD as our feature extractor[1]. Here, we adopt the state-of-the-art methods that only use RGB information for comparison. The results are shown in Table II. CF-GNN again outperforms other state-of-the-art methods in all evaluation metrics. We notice that the improvement in video-to-text retrieval is better than in text-to-video retrieval on both datasets. We believe this is because in video-to-text retrieval, a video usually has more than one relevant text which alleviates the class imbalance problem.

We also evaluate our proposed model on the TGIF dataset as we believe TGIF is more challenging for video-text retrieval. Each GIF in the TGIF training set is annotated with only one

[1] Dual Encoding [7] does not provide the pretrained model on MSVD and TGIF.

TABLE IV: **Image-text retrieval on MS-COCO.** The results show the generalization ability of our model.

Method	Text-to-Image Retrieval						Image-to-Text Retrieval					
	R@1	R@5	R@10	Med r	Mean r	mAP	R@1	R@5	R@10	Med r	Mean r	mAP
VSE++[49]	33.7	68.8	81.0	**3**	12.9	0.492	43.6	74.8	84.6	**2**	8.8	0.387
Ours + VSE++	**35.8**	**70.1**	**81.6**	**3**	**12.5**	**0.507**	**46.0**	**76.3**	**85.3**	**2**	**8.5**	**0.394**

TABLE V: **Ablation on MSR-VTT.** The results show that all the components in our framework are useful.

Method	Text-to-Video Retrieval						Video-to-Text Retrieval					
	R@1	R@5	R@10	Med r	Mean r	mAP	R@1	R@5	R@10	Med r	Mean r	mAP
1 layer	7.8	22.8	32.2	32	206.3	0.158	13.2	31.9	43.0	16	122.2	0.067
2 layers	7.9	23.0	32.3	32	206.2	0.158	13.7	32.1	43.9	15	121.9	0.068
w/o SL	7.9	22.8	32.0	32	206.2	0.159	14.0	31.9	43.5	16	122.2	0.067
w/o EC	7.7	22.3	32.0	32	206.4	0.158	13.7	31.7	43.5	16	122.3	0.067
w/o SM	7.8	22.4	31.8	32	206.6	0.156	13.6	**32.2**	43.3	15	122.3	0.067
w/o EI	7.9	22.6	31.9	32	206.6	0.157	13.4	32.1	43.1	16	122.4	0.066
w/o C2F	7.9	22.9	32.4	32	206.2	0.156	13.6	31.9	43.9	15	122.0	0.067
Full (CL)	7.7	22.8	32.3	32	206.2	0.156	13.7	**32.2**	43.4	15	122.2	0.067
Full (FL)	**8.0**	22.9	32.5	**31**	206.1	**0.160**	13.6	32.1	43.4	15	122.0	0.067
Full	**8.0**	**23.2**	**32.6**	**31**	**206.0**	**0.160**	**14.3**	**32.2**	**44.3**	**14**	**121.3**	**0.069**

TABLE VI: **Coarse-to-fine parameter k^l.** The table shows different versions of $\{k^l\}$ set.

Method	Text-to-Video Retrieval						Video-to-Text Retrieval					
	R@1	R@5	R@10	Med r	Mean r	mAP	R@1	R@5	R@10	Med r	Mean r	mAP
k^l=40,30,10	23.8	55.7	69.7	4	20.4	0.386	37.0	66.9	78.1	3	11.3	0.240
k^l=60,40,20	24.1	56.0	70.1	4	19.9	0.389	37.4	67.3	78.8	3	11.0	0.249
k^l=80,60,40	24.2	56.4	70.4	4	20.3	0.390	37.4	67.3	79.0	3	11.1	0.249
k^l=90,70,50	24.2	56.3	70.0	4	20.6	0.389	37.8	68.6	79.4	2	11.1	0.246
k^l=90,60,30	24.3	56.6	70.4	4	20.4	0.392	37.9	68.1	79.1	2	11.1	0.246

sentence while in validation and testing set it is annotated with 3 sentences, which brings a severe class imbalance in video-text retrieval. Besides, the lack of descriptions for videos further aggravates the difficulty in exploiting structural similarities by CF-GNN. Consequently, compared with other two datasets, our CF-GNN does not obtain significant improvements in TGIF. Even so, as shown in Table III, our method still outperforms other state-of-the-arts[2], which shows the effectiveness of our approach.

We also noticed another paper HGR [41] for video-text retrieval, which was proposed at almost the same time and obtained a higher performance. The relations with HGR have been clarified in Related Work (see Section II). Note that our model mainly focuses on the structural modeling and relationship refinement among videos and texts rather than learning better feature representations. Of course we can further boost our model's performance as long as we adopt the feature extractor in HGR and exploit structural similarities with our CF-GNN, but it is unnecessary indeed.

Image-Text Retrieval. To evaluate the cross-domain generalization of our proposed model, we perform image-text retrieval on MS-COCO dataset. We retrain the VSE++ [49] model on MS-COCO as feature extractor, following the setting in [7]. The results are reported by averaging over 5 folds of 1,000 test images. As shown in Table IV, CF-GNN outperforms VSE++ in all evaluation metrics, suggesting a good cross-domain generalization of our model. Only one method is compared because our model mainly focuses on video-text retrieval and we perform image-text retrieval experiment based on the popular VSE++ just in order to evaluate CF-GNN's generalization ability.

TABLE VII: **Sum of recall in each layer on three datasets.**

	MSR-VTT	MSVD	TGIF
1-layer	150.8	280.9	142.8
2-layer	152.9	284.2	144.0
3-layer	154.6	286.1	145.2

D. Further Remarks

To investigate the effectiveness of all the components proposed in our model, we conduct a series of ablation studies on MSR-VTT dataset using the pretrained feature extractor in Dual Encoding [7]. Every time we remove or modify one

[2] We cite the results of the state-of-the-arts (except Dual Encoding) on TGIF from the second version of [54] on arXiv.

component from the full model and check the corresponding performance. We fix the settings of our model to ensure the results are comparable. Table V shows the corresponding results of several ablated versions of CF-GNN.

1 layer. The ablated version of 1 layer constructs a one-layer CF-GNN that cannot perform multi-step coarse-to-fine video-text retrieval, which shows lower performance.

2 layers. This ablated version constructs a two-step coarse-to-fine video-text retrieval framework. Compared with the ablated version of 1 layer, the increase of the overall performance indicates the effectiveness of multi-step coarse-to-fine video-text retrieval.

W/o SL. In this version, we omit the structural context-aware loss to see if it is beneficial. From the results we can observe its effectiveness.

W/o EC. This version aims to study whether the edge feature concatenation is useful. Dimensions of edge features in every layer are 3 in this version. Results show a significant drop in retrieval performance, meaning that edge feature concatenation plays an important role in CF-GNN.

W/o SM. For this version, we check whether the samples from the same modality of the query (*i.e.* the top-K most similar texts to the query in text-to-video retrieval) are helpful. As shown in Table V, samples from the same modality of the query can provide contextual information for cross-modal retrieval.

W/o EI. To evaluate whether the heuristic information used in edge feature initialization works in CF-GNN, we randomly initialize edge features with values between 0 and 1 and the accuracy drops.

W/o C2F. We disable the coarse-to-fine strategy in our model and do not refine the candidate videos/texts every layer. The results prove the effectiveness of our coarse-to-fine retrieval strategy.

Full (CL). Here we replace our proposed random-drop focal loss with the widely-used cross entropy loss. Decrease can be seen in performance, which are mainly accused of the class imbalance issue.

Full (FL). This version utilizes the commonly used focal loss [57] for model training. From Table V we can observe that CF-GNN with random-drop focal loss obtains a better performance than the original focal loss.

Coarse-to-Fine Parameter k^l. We investigate the influence of the parameter k^l in each layer of our coarse-to-fine video-text retrieval framework. We test several sets of $\{k^l\}$ while keeping other parameters fixed. The results on MSR-VTT using pretrained feature extractor CE [8] are presented in Table VI. Generally speaking, our model is robust to the parameter k^l in each layer, as long as they are well-proportioned between 0 and 100. Parameter sets $\{k^l\}$ with extreme proportions will lead to a decrease in retrieval performance (*i.e.* $k^l = 40, 30, 10$).

Corase-to-Fine Results in Each Layer. We take out the text-to-video and video-to-text retrieval results in each layer of CF-GNN and calculate the sum of recall@1,5,10. We use Dual Encoding [7] as the feature extractor. As shown in Table VII, the performance of CF-GNN increases layer-by-layer, indicating the effectiveness of our coarse-to-fine framework.

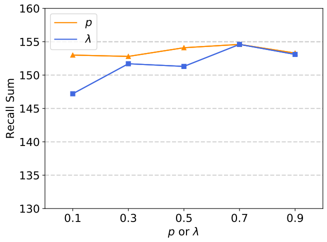


Fig. 3: Parameter analysis with different p and λ

λ in loss function. The trade-off hyperparameter λ in our loss function is also investigated. As shown in Figure 3, we vary λ from 0.1 to 0.9 and calculate the sum of recall in video-text retrieval on MSR-VTT using feature extractor in Dual Encoding [7]. We can observe that the accuracy is quite low when λ is 0.1, which means we focus too much on the structural context-aware loss. The accuracy also drops when λ is 0.9 and achieves its highest value at $\lambda = 0.7$, indicating that a good combination of our proposed random-drop focal loss and structural context-aware loss is beneficial to our model.

p in loss function. We investigate the sensitivity of the parameter p in loss function. As shown in Figure 3, we vary p from 0.1 to 0.9 and calculate the sum of recall in video-text retrieval on MSR-VTT using feature extractor in Dual Encoding [7]. Results indicate that dropping too many negative samples or preserving most negative samples will decrease the performance. A proper random-drop strategy is useful to CF-GNN.

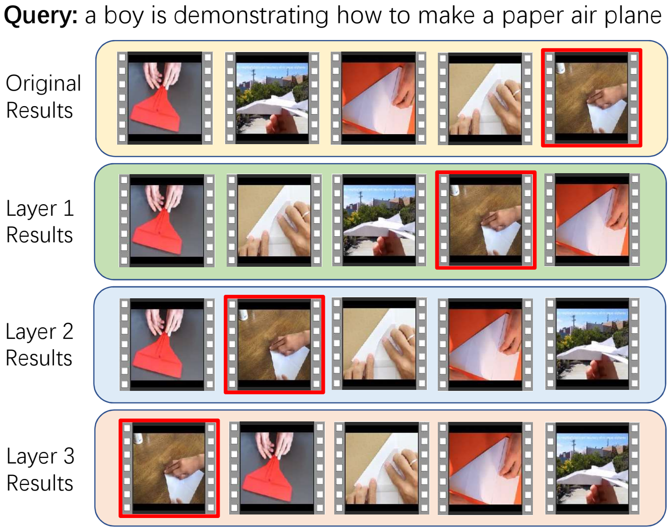
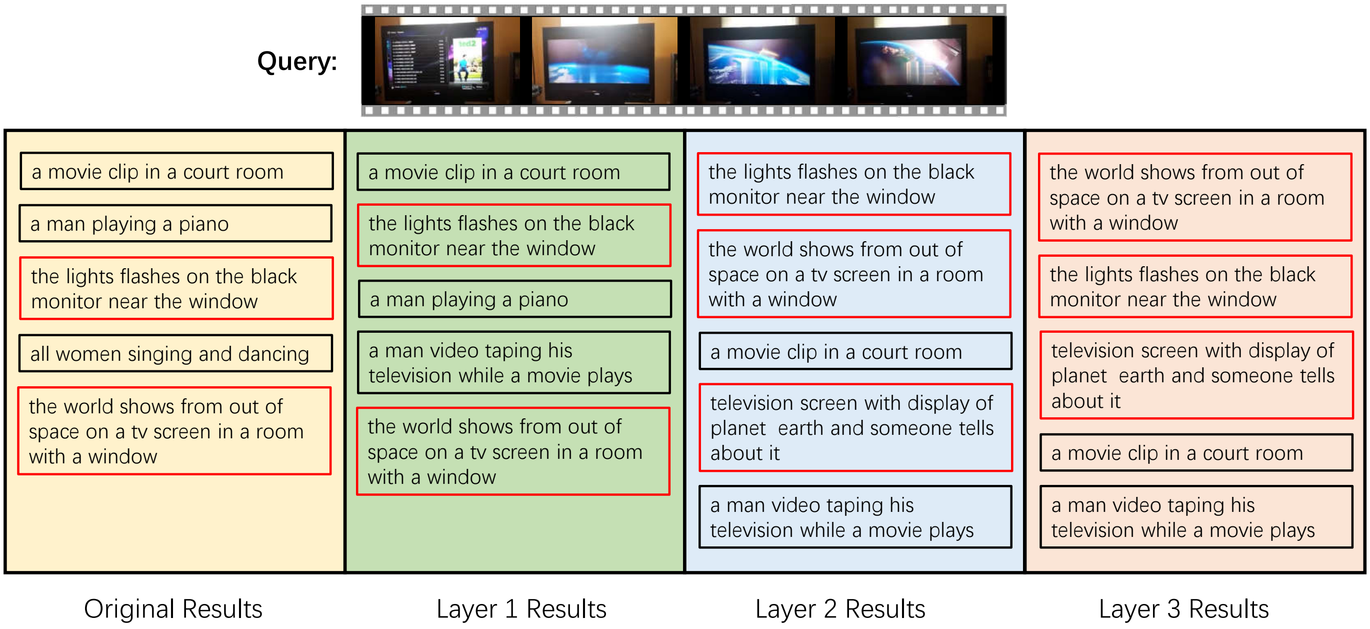


Fig. 4: Examples of the top-5 text-to-video retrieval results in each layer of CF-GNN on MSR-VTT dataset. We employ the pretrained feature extractor CE [8]. The video with a red bounding box is the ground truth.

E. Qualitative Analysis

We present some qualitative examples of the top-5 video-text retrieval results in each layer of CF-GNN on MSR-VTT



Original Results	Layer 1 Results	Layer 2 Results	Layer 3 Results
a movie clip in a court room	a movie clip in a court room	the lights flashes on the black monitor near the window	the world shows from out of space on a tv screen in a room with a window
a man playing a piano	the lights flashes on the black monitor near the window	the world shows from out of space on a tv screen in a room with a window	the lights flashes on the black monitor near the window
the lights flashes on the black monitor near the window	a man playing a piano	a movie clip in a court room	television screen with display of planet earth and someone tells about it
all women singing and dancing	a man video taping his television while a movie plays	television screen with display of planet earth and someone tells about it	a movie clip in a court room
the world shows from out of space on a tv screen in a room with a window	the world shows from out of space on a tv screen in a room with a window	a man video taping his television while a movie plays	a man video taping his television while a movie plays

Fig. 5: Examples of the top-5 video-to-text retrieval results in each layer of CF-GNN on MSR-VTT dataset. We employ the pretrained feature extractor CE [8]. The text with a red bounding box is the ground truth. Top-ranked texts are at the top of the table.

datasest. Our model retrieves the candidate videos/texts in a coarse-to-fine manner by exploiting structural similarities among video and text data points in common space. As shown in Figure 4 and Figure 5, the original results are from the base model (CE [8]) and the ground-truth video/text with a red bounding box is gradually ranked to the top via multi-step reasoning. Since the candidate videos/texts are quite similar and easy-confused, they may not be well retrieved in previous methods. CF-GNN exploits latent semantic relations among all top-K videos and top-K texts to the query and retrieves videos/texts in a coarse-to-fine manner, which gradually distinguishes the subtle semantic differences among them, thus improving the retrieval performance.

F. Efficiency Test

In terms of inference, to save computational overhead, we construct the graph with only the top-K (K=100) most similar samples as vertices. The computational complexity of selecting the top-K from N samples based on min-heap is $O(N)$ (when $K \ll N$). Base models (*e.g.* Dual Encoding [7], Collaborative Experts [8]) find out the closest sample in common space as the retrieval result, whose minimum computational complexity is also $O(N)$. Therefore, the extra time overhead in our model mostly lies in message passing in GNNs, which can be efficiently executed by GPUs. Specifically, with a TITAN RTX GPU, the average inference time of our model and Dual Encoding [7] (base model) are 0.09 and 0.06 seconds per query respectively on the MSR-VTT dataset. When it comes to training, with the help of extracted video and text features, our model takes about 8 hours to achieve its convergence on the MSR-VTT dataset. Note that once the network is trained, the inference can be performed independently, which means we can process large-scale training offline and respond to queries on the fly.

V. Conclusion

In this paper, we propose a novel coarse-to-fine graph neural network for video-text retrieval which progressively improves the model discriminative ability and locates the positive sample via multi-step reasoning. To the best of our knowledge, we are among the first to use GNNs to model structural similarities among data points in common space and perform video-text retrieval with a coarse-to-fine strategy. We focus on exploiting edge features in GNNs and initialize them with heuristic information. By designing the learnable scoring modules that refine the retrieval results layer by layer and a random-drop focal loss that alleviates class imbalance and impact of noise samples, CF-GNN jointly enjoys the merits of highly discriminative ability, robust relationship refinement and balanced training of positive and negative samples. Our model outperforms the state-of-the-art methods on three popular video-text retrieval benchmarks. The results of ablation study prove the effectiveness of every component in our model. Besides, the favorable results on MS-COCO dataset for image-text retrieval consistently demonstrate the promising potential of CF-GNN.

As the first step of using GNNs for video-text retrieval, we put more concentration on the structural modeling than video/text representation learning. In the future, we plan to make better use of the temporal/scene/object/action/knowledge information of videos and texts in an end-to-end manner. Besides, the proposed framework has great potential to be applied for other tasks such as cross-modal hashing and person re-identification, which can be further explored.

References

[1] N. C. Mithun, J. Li, F. Metze, and A. K. Roy-Chowdhury, "Learning joint embedding with multimodal cues for cross-modal video-text retrieval," in *Proceedings of the 2018 ACM on International Conference on Multimedia Retrieval*. ACM, 2018, pp. 19–27.

[2] X. Yang, T. Zhang, and C. Xu, "Semantic feature mining for video event understanding," *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, vol. 12, no. 4, p. 55, 2016.

[3] X. Yang, T. Zhang, C. Xu, and M. S. Hossain, "Automatic visual concept learning for social event understanding," *IEEE Transactions on Multimedia*, vol. 17, no. 3, pp. 346–358, 2015.

[4] R. Xu, C. Xiong, W. Chen, and J. J. Corso, "Jointly modeling deep video and compositional text to bridge vision and language in a unified framework," in *Twenty-Ninth AAAI Conference on Artificial Intelligence*, 2015.

[5] Y. Pan, T. Mei, T. Yao, H. Li, and Y. Rui, "Jointly modeling embedding and translation to bridge video and language," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 4594–4602.

[6] A. Frome, G. S. Corrado, J. Shlens, S. Bengio, J. Dean, M. Ranzato, and T. Mikolov, "Devise: A deep visual-semantic embedding model," in *Advances in neural information processing systems*, 2013, pp. 2121–2129.

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

[8] Y. Liu, S. Albanie, A. Nagrani, and A. Zisserman, "Use what you have: Video retrieval using representations from collaborative experts," in *British Machine Vision Conference*.

[9] X. Wang, Y. Hua, E. Kodirov, G. Hu, R. Garnier, and N. M. Robertson, "Ranked list loss for deep metric learning," 2019.

[10] K. Sohn, "Improved deep metric learning with multi-class n-pair loss objective," in *Advances in Neural Information Processing Systems*, 2016, pp. 1857–1865.

[11] J. Gao, T. Zhang, and C. Xu, "Learning to model relationships for zero-shot video classification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.

[12] M. Prates, P. H. Avelar, H. Lemos, L. C. Lamb, and M. Y. Vardi, "Learning to solve np-complete problems: A graph neural network for decision tsp," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, 2019, pp. 4731–4738.

[13] J. Gao, T. Zhang, and C. Xu, "I know the relationships: Zero-shot action recognition via two-stream graph convolutional networks and knowledge graphs," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, 2019, pp. 8303–8311.

[14] ——, "Graph convolutional tracking," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 4649–4659.

[15] J. Gao and C. Xu, "Ci-gnn: Building a category-instance graph for zero-shot video classification," *IEEE Transactions on Multimedia*, 2020.

[16] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," *arXiv preprint arXiv:1609.02907*, 2016.

[17] L. Gong and Q. Cheng, "Exploiting edge features for graph neural networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 9211–9219.

[18] J. Kim, T. Kim, S. Kim, and C. D. Yoo, "Edge-labeling graph neural network for few-shot learning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 11–20.

[19] M. Schlichtkrull, T. N. Kipf, P. Bloem, R. Van Den Berg, I. Titov, and M. Welling, "Modeling relational data with graph convolutional networks," in *European Semantic Web Conference*. Springer, 2018, pp. 593–607.

[20] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *European conference on computer vision*. Springer, 2014, pp. 740–755.

[21] Y. Hao, T. Mu, R. Hong, M. Wang, N. An, and J. Y. Goulermas, "Stochastic multiview hashing for large-scale near-duplicate video retrieval," *IEEE Transactions on Multimedia*, vol. 19, no. 1, pp. 1–14, 2016.

[22] W. Jing, X. Nie, C. Cui, X. Xi, G. Yang, and Y. Yin, "Global-view hashing: harnessing global relations in near-duplicate video retrieval," *World Wide Web*, vol. 22, no. 2, pp. 771–789, 2019.

[23] J. Song, Y. Yang, Z. Huang, H. T. Shen, and R. Hong, "Multiple feature hashing for real-time large scale near-duplicate video retrieval," in *Proceedings of the 19th ACM international conference on Multimedia*. ACM, 2011, pp. 423–432.

[24] G. Zhao and M. Pietikainen, "Dynamic texture recognition using local binary patterns with an application to facial expressions," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, no. 6, pp. 915–928, 2007.

[25] Y.-G. Jiang, C.-W. Ngo, and J. Yang, "Towards optimal bag-of-features for object categorization and semantic video retrieval," in *Proceedings of the 6th ACM international conference on Image and video retrieval*. ACM, 2007, pp. 494–501.

[26] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, vol. 60, no. 2, pp. 91–110, 2004.

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

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

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

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

[31] Y.-G. Jiang, J. Wang, Q. Wang, W. Liu, and C.-W. Ngo, "Hierarchical visualization of video search results for topic-based browsing," *IEEE Transactions on Multimedia*, vol. 18, no. 11, pp. 2161–2170, 2016.

[32] X. Wu, A. G. Hauptmann, and C.-W. Ngo, "Practical elimination of near-duplicates from web video search," in *Proceedings of the 15th ACM international conference on Multimedia*. ACM, 2007, pp. 218–227.

[33] Y. Cai, L. Yang, W. Ping, F. Wang, T. Mei, X.-S. Hua, and S. Li, "Million-scale near-duplicate video retrieval system." in *ACM Multimedia*, 2011, pp. 837–838.

[34] R. Xu, C. Li, J. Yan, C. Deng, and X. Liu, "Graph convolutional network hashing for cross-modal retrieval," in *Proceedings of the 28th International Joint Conference on Artificial Intelligence*. AAAI Press, 2019, pp. 982–988.

[35] X. Ma, T. Zhang, and C. Xu, "Multi-level correlation adversarial hashing for cross-modal retrieval," *IEEE Transactions on Multimedia*, 2020.

[36] X. Lu, L. Zhu, J. Li, H. Zhang, and H. T. Shen, "Efficient supervised discrete multi-view hashing for large-scale multimedia search," *IEEE Transactions on Multimedia*, 2019.

[37] F. Scarselli, M. Gori, A. C. Tsoi, M. Hagenbuchner, and G. Monfardini, "The graph neural network model," *IEEE Transactions on Neural Networks*, vol. 20, no. 1, pp. 61–80, 2008.

[38] V. Garcia and J. Bruna, "Few-shot learning with graph neural

networks," *arXiv preprint arXiv:1711.04043*, 2017.

[39] J. Gao, T. Zhang, and C. Xu, "Watch, think and attend: End-to-end video classification via dynamic knowledge evolution modeling," in *Proceedings of the 26th ACM international conference on Multimedia*, 2018, pp. 690–699.

[40] T. Kipf, E. Fetaya, K.-C. Wang, M. Welling, and R. Zemel, "Neural relational inference for interacting systems," *arXiv preprint arXiv:1802.04687*, 2018.

[41] S. Chen, Y. Zhao, Q. Jin, and Q. Wu, "Fine-grained video-text retrieval with hierarchical graph reasoning," *arXiv preprint arXiv:2003.00392*, 2020.

[42] J. Yu, Y. Lu, Z. Qin, W. Zhang, Y. Liu, J. Tan, and L. Guo, "Modeling text with graph convolutional network for cross-modal information retrieval," in *Pacific Rim Conference on Multimedia*. Springer, 2018, pp. 223–234.

[43] C. Chang, G. Yu, C. Liu, and M. Volkovs, "Explore-exploit graph traversal for image retrieval," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 9423–9431.

[44] F. Magliani, K. McGuinness, E. Mohedano, and A. Prati, "An efficient approximate knn graph method for diffusion on image retrieval," *arXiv preprint arXiv:1904.08668*, 2019.

[45] J. Zhang, Y. Peng, and M. Yuan, "Unsupervised generative adversarial cross-modal hashing," in *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

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

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

[48] Y. Li, Y. Song, L. Cao, J. Tetreault, L. Goldberg, A. Jaimes, and J. Luo, "TGIF: A New Dataset and Benchmark on Animated GIF Description," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[49] F. Faghri, D. J. Fleet, J. R. Kiros, and S. Fidler, "Vse++: Improved visual-semantic embeddings," *arXiv preprint arXiv:1707.05612*, vol. 2, no. 7, p. 8, 2017.

[50] S. Venugopalan, M. Rohrbach, J. Donahue, R. Mooney, T. Darrell, and K. Saenko, "Sequence to sequence-video to text," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 4534–4542.

[51] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, "Automatic differentiation in pytorch," 2017.

[52] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[53] R. Socher and L. Fei-Fei, "Connecting modalities: Semi-supervised segmentation and annotation of images using unaligned text corpora," in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. IEEE, 2010, pp. 966–973.

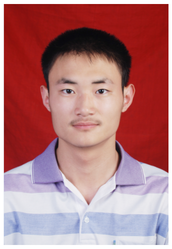
[54] Y. Song and M. Soleymani, "Polysemous visual-semantic embedding for cross-modal retrieval," 2019.

[55] I. Vendrov, R. Kiros, S. Fidler, and R. Urtasun, "Order-embeddings of images and language," *arXiv preprint arXiv:1511.06361*, 2015.

[56] F. Feng, X. Wang, and R. Li, "Cross-modal retrieval with correspondence autoencoder," in *Proceedings of the 22nd ACM international conference on Multimedia*. ACM, 2014, pp. 7–16.

[57] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2980–2988.

Wei Wang received the bachelor's degree in electronic and information engineering from Chongqing University, Chongqing, China, in 2018. He is currently a Ph.D candidate at the Multimedia Computing Group, National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing, China. His research interests include multimedia analysis and computer vision, especially deep learning, multimedia computing and video understanding.



Junyu Gao received the bachelor's degree in computer science from Xi'an JiaoTong University, Xi'an, Shaanxi, China, in 2015, and the Ph.D. degree in pattern recognition and intelligent systems from the Institute of Automation, Chinese Academy of Sciences, Beijing, China, in 2020. Currently, he is an Assistant Professor at the Institute of Automation, Chinese Academy of Sciences, Beijing, China. His research interests include multimedia analysis and computer vision, especially video understanding, deep learning, and cross-modal retrieval.

Xiaoshan Yang received the master's degree in computer science from Beijing Institute of Technology, Beijing, China, in 2012, and the Ph.D. degree in pattern recognition and intelligent systems from the Institute of Automation, Chinese Academy of Sciences, Beijing, China, in 2016. Currently, he is an Associate Professor at the Institute of Automation, Chinese Academy of Sciences, Beijing, China. His research interests include multimedia analysis and computer vision.

Changsheng Xu (M'97–SM'99–F'14) is a Distinguished Professor in National Lab of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences. His research interests include multimedia content analysis/indexing/retrieval, pattern recognition and computer vision. He has hold 40 granted/pending patents and published over 300 refereed research papers in these areas. Dr. Xu has served as associate editor, guest editor, general chair, program chair, area/track chair, special session organizer, session chair and TPC member for over 20 IEEE and ACM prestigious multimedia journals, conferences and workshops, including IEEE Trans. on Multimedia, ACM Trans. on Multimedia Computing, Communications and Applications and ACM Multimedia conference. He is IEEE Fellow, IAPR Fellow and ACM Distinguished Scientist.