Hierarchical Cross-Modal Graph Consistency Learning for Video-Text Retrieval

Weike Jin
Zhejiang University, Hangzhou
weikejin@zju.edu.cn

Zhou Zhao*
Zhejiang University, Hangzhou
zhaozhou@zju.edu.cn

Pengcheng Zhang
Zhejiang University, Hangzhou
zhangpengcheng1218@zju.edu.cn

Jieming Zhu
Huawei Noah's Ark Lab, Shenzhen
jamie.zhu@huawei.com

Xiuqiang He
Huawei Noah's Ark Lab, Shenzhen
hexiuqiang1@huawei.com

Yueting Zhuang
Zhejiang University, Hangzhou
yzhuang@zju.edu.cn

ABSTRACT

Due to the popularity of video contents on the Internet, the information retrieval between videos and texts has attracted broad interest from researchers, which is a challenging cross-modal retrieval task. A common solution is to learn a joint embedding space to measure the cross-modal similarity. However, many existing approaches either pay more attention to textual information, video information, or cross-modal matching methods, but less to all three. We believe that a good video-text retrieval system should take into account all three points, fully exploiting the semantic information of both modalities and considering a comprehensive match. In this paper, we propose a Hierarchical Cross-Modal Graph Consistency Learning Network (HCGC) for video-text retrieval task, which considers multi-level graph consistency for video-text matching. Specifically, we first construct a hierarchical graph representation for the video, which includes three levels from global to local: video, clips and objects. Similarly, the corresponding text graph is constructed according to the semantic relationships among sentence, actions and entities. Then, in order to learn a better match between the video and text graph, we design three types of graph consistency (both direct and indirect): inter-graph parallel consistency, inter-graph cross consistency and intra-graph cross consistency. Extensive experimental results on different video-text datasets demonstrate the effectiveness of our approach on both text-to-video and video-to-text retrieval.

CCS CONCEPTS

• **Information systems** → **Multimedia and multimodal retrieval**; **Video search**.

KEYWORDS

cross-modal learning, video-text retrieval, graph consistency

*Zhou Zhao is the corresponding author.

SIGIR '21, July 11–15, 2021, Virtual Event, Canada.

ACM ISBN 978-1-4503-8037-9/21/07…$15.00
https://doi.org/10.1145/3404835.3462974

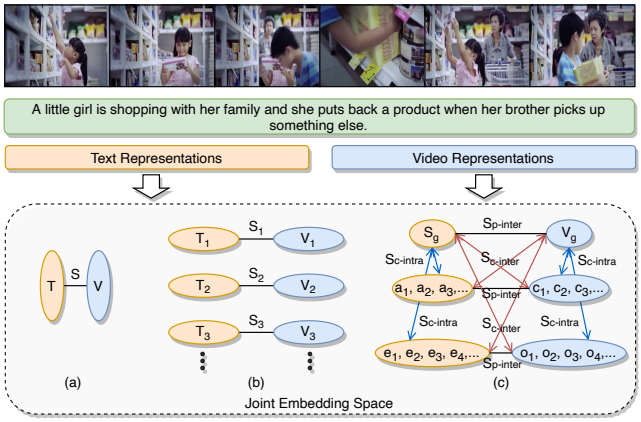

Figure 1: Different cross-modal matching schemes. (a) shows the basic single vector-based similarity. (b) is a normal multi-level representation matching. (c) is our hierarchical cross-modal graph consistency learning scheme.

ACM Reference Format:
Weike Jin, Zhou Zhao, Pengcheng Zhang, Jieming Zhu, Xiuqiang He, and Yueting Zhuang. 2021. Hierarchical Cross-Modal Graph Consistency Learning for Video-Text Retrieval. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '21), July 11–15, 2021, Virtual Event, Canada.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3404835.3462974

1 INTRODUCTION

Due to the rapid development of Internet and communication technology, today's presentation of information is more diversified than before. Especially for video media, it has become increasingly popular and ubiquitous. Long-form video websites like YouTube and short-form video applications like TikTok generate huge amounts of video data every day. How to make full use of this data has become an important research topic. Video-text retrieval [2, 8, 37] is one of the fundamental tasks, which has broad applications such as search engine, intelligent editing and content recommendation [50]. It can be divided into two types: retrieving the video according to users' requirements in natural language (text-to-video) and retrieving the text that best matches the video content (video-to-text).

Since video and text are heterogeneous information, single modality approaches cannot be directly applied to video-text retrieval. In

order to bridge the gap between these two modalities, early methods [2, 6, 16, 49] transform the video content into a set of predefined textual descriptions named visual concepts. And the text is also transformed into a set of concepts, then, single modality matching can be used between the visual and textual concepts. However, there are two problems. One is that the selection of accurate concepts can be challenging. Even if the concepts are classified accurately, they may not be able to adequately represent the temporal context of the video and complex semantics of the text. Due to the limitations of concepts, word embeddings [30] and visual features [21] methods are utilized to learn a better information representation and a joint embedding space is used to measure the cross-modal similarity [12, 31, 46].

Many existing approaches either pay more attention to textual information, video information, or cross-modal matching methods, but less to all three. A common solution is to utilize recurrent neural networks (RNNs) to learn a dense vector representation for the natural language sentence. As for the video, convolutional neural networks (CNNs) are employed to extract frame-level or clip-level features, then the final video representation can be obtained by temporal aggregation. Such video and text vectors can be regarded as global representations, which are mapped into a joint embedding space, and the semantic similarity is calculated between them, as shown in Figure 1(a). When encountered with complex sentence, a simple global vector may not adequately represent the information. Lin et al. [25] use a structured semantic form to represent the sentence based on the syntactic annotations. Recently, Chen et al. [4] disentangle text into a hierarchical semantic graph including three levels of events, actions, entities, and generate hierarchical textual embeddings through attention-based graph reasoning. They also add a local-level video-text matching besides the global-level matching, as shown in Figure 1(b). Since videos have a complex spatial-temporal structure, improvements can also be achieved in this domain. Dong et al. [8] encode frame-level video features in a multi-level strategy for learning powerful dense representations. Feng et al. [11] propose a visual semantic enhanced reasoning network to generate visual representations by exploiting object relations. The enhanced frame representations capture vital regions and suppress redundancy in a scene. Although these approaches have achieved performance improvements in different aspects, learning semantic alignments between text and video is still a challenging problem.

We believe that a good video-text retrieval system should take into account all three points, exploiting the semantic information of both video and text while considering a comprehensive match. Towards this goal, in this paper, we propose a Hierarchical Cross-Modal Graph Consistency Learning Network (HCGC) for video-text retrieval task, which considers multi-level graph consistency for cross-modal matching based on the fine-grained structured video-text graph representation. Specifically, we first construct a hierarchical graph representation for the video, which includes three levels from global to local: video, clips and objects. The global level representation of video is extracted by a temporal aggregation module. The video clips are first segmented by temporal segmentation algorithm [34] according to the shot boundaries and semantic boundaries. Then, the clips are densely connected and a graph convolutional network is utilized to learn potential semantic relationships between different clips. For each clip, the object level representations are obtained via an attention-based aggregation from the regions of frames. Similarly, the corresponding text graph is constructed according to the semantic relationships among sentence, actions and entities level. Then, in order to learn a better match between different layers of the video and text graphs, we design three types of graph consistency (both direct and indirect): inter-graph parallel consistency (learning the direct consistency between the corresponding layers of different modal graphs), inter-graph cross consistency (learning indirect consistency between different layers of different modal graphs) and intra-graph cross consistency (learning indirect consistency between different layers of the same modal graph), as shown in Figure 1(c). Extensive experimental results on different video-text retrieval datasets demonstrate the effectiveness of our approach on both text-to-video and video-to-text retrieval. The main contributions of this work are summarized as follows:

- We propose a hierarchical cross-modal graph consistency learning network (HCGC) for video-text retrieval task, which constructs hierarchical graph representations for both the video and text, and utilizes multi-level graph consistency to help with cross-modal matching.
- We design three types of graph consistency between different layers of the video and text graph (both direct and indirect): inter-graph parallel consistency, inter-graph cross consistency and intra-graph cross consistency, to learn a better cross-modal matching.
- We conduct extensive experiments on different datasets to demonstrate that our approach can achieve better performance on both text-to-video and video-to-text retrieval.

2 RELATED WORKS

Vision-language research has become a popular research area, which requires an understanding of visual contents, language semantics and relationships between them. Video-text retrieval is one of the basic tasks. Though image-text retrieval [9, 12–14, 19, 26, 28, 35, 39, 44] has been widely explored, video-text retrieval is still quite challenging because videos contain multi-modality information and spatial-temporal characteristics. The majority of the previous methods can be roughly divided into two types: concept-based and embedding-based methods. Most of the early methods are concept-based, which generally use visual concept classifiers to describe video content and linguistic rules to detect textual concepts. For example, Le et al. [22] use bag-of-visual word model with geometric verification to search for shots with the query location. Markatopoulou et al. [29] present a set of steps to cleverly analyse different parts of the query in order to convert it to related semantic concepts, then the similarity between a given query and a specific video is measured by concept matching. Ueki et al. [41] construct a much larger concept bank that includes more than 50k concepts, and in addition to the pre-trained CNN model, they train a support vector machine (SVM) classifier to annotate video content automatically. Tao et al. [40] propose a concept learning method to crawl related images by image search engine, and train a SVM model to classify concepts which are not involved in the concept

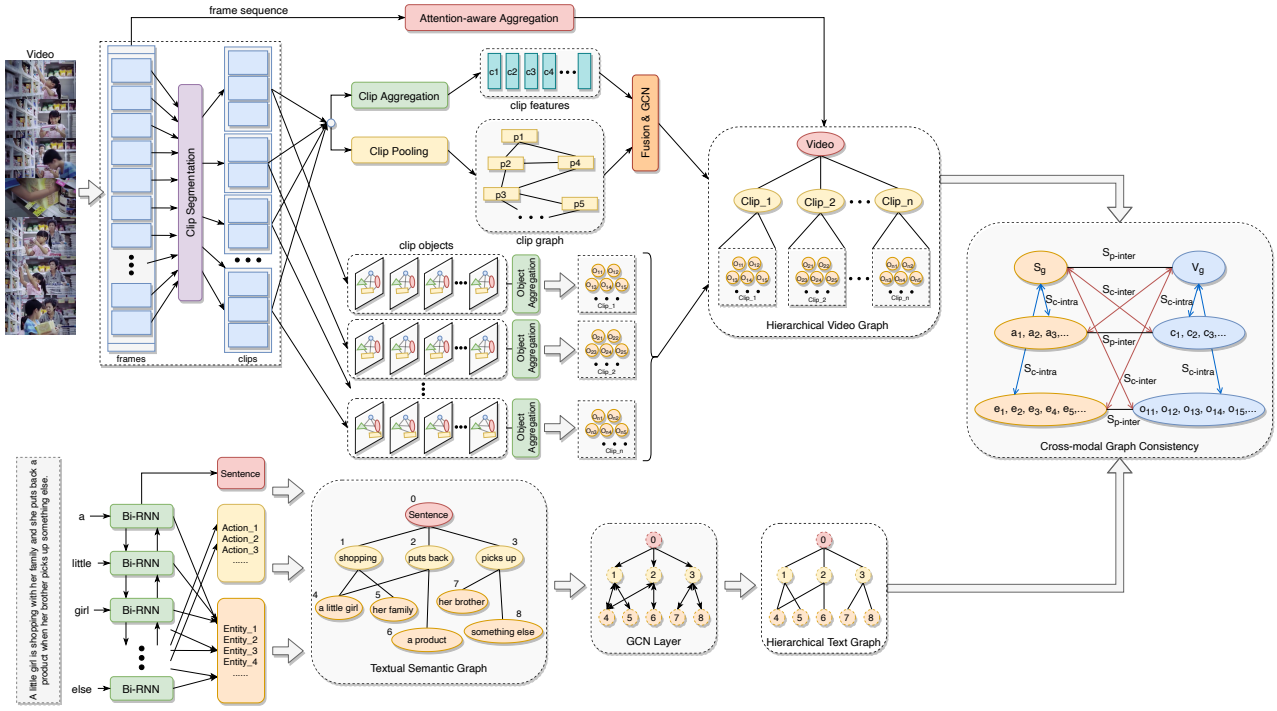

Figure 2: An overview of our hierarchical cross-modal graph consistency learning network (HCGC) for video-text retrieval.

library. Although some progress has been made, it's still difficult for the concept-based methods to fully explore the diversity and contextual relevance of texts and videos within limited concepts.

Due to the rapid development of deep neural network, more concept free methods are proposed, which directly encode the video and text into a common space and then semantic matching is realized in the common subspace. Different approaches focus on different parts of the embedding-based pipeline. For text encoding, Habibian et al [17] propose an embedding between the video features and their textual descriptions, which is learned by utilizing the correlations between the words in the descriptions. They utilize bag-of-words model to encode the video descriptions. Different variants of recurrent neural networks (LSTM, GRU, bidirectional RNN, etc.) are the most widely used sentence encoder [31, 32, 43]. Li et al. [23] make a further step by using multiple text embedding strategies including bag-of-words, word2vec, and GRU to learn a robust text representation. Recently, more complex structured representations are used for text encoding. Dong et al. [8] utilize a multi-level text encoding to capture the global, local, and sequential patterns from the text. Yang er al. [47] propose a Tree-augmented Cross-modal Encoding method by jointly learning the linguistic structure of queries and the temporal representation of videos. Chen et al. [4] propose a Hierarchical Graph Reasoning (HGR) model, which decomposes video-text matching into global-to-local levels. The model disentangles text into a hierarchical semantic graph including three levels of events, actions, entities, and generates hierarchical textual embeddings via attention-based graph reasoning. However, their video encoding is straightforward, which may not take full advantage of the hierarchical text graph.

As for video encoding, a common solution is employing convolutional neural networks (CNNs) to extract frame-level or clip-level features, then the final video representation can be obtained by temporal aggregation [7, 31]. Mithun et al. [31] simultaneously utilize multi-modal features (different visual characteristics, audio inputs, and text) by a fusion strategy for efficient retrieval. Liu et al. [27] propose a collaborative experts model to aggregate information from multimodal cues such as image, motion and audio for video encoding. Dong et al. [8] process video and text separately and utilize bi-GRU, CNN and mean pooling for three-level encoding on both sides, in order to explicitly and progressively exploit global, local and temporal patterns features. Feng et al. [11] propose a visual semantic enhanced reasoning network to generate visual representations by exploiting object relations. Yang et al. [47] jointly model the temporal dependence between frames and frame-wise temporal interaction in the temporal attentive video encoder, followed by an attentive pooling mechanism to vectorize the video. Yu et al. [48] propose a joint sequence fusion (JSFusion) model, which leverages recursively learnable attention modules for measuring semantic matching scores between multimodal sequence data. Song et al. [38] propose a polysemous instance embedding network (PIE-Net) which learns multiple and diverse representations per instance. Different from the previous work, we construct a hierarchical video graph for fine-grained structured video representations.

3 METHOD

Figure 2 gives an overview of our hierarchical cross-modal graph consistency learning network (HCGC) for video-text retrieval, which

consists of three major parts: 1) hierarchical text graph module; 2) hierarchical video graph module; and 3) multi-level cross-modal graph consistency learning module.

3.1 Hierarchical Text Graph

Transforming text into semantic graphs has been widely explored, here, we simply follow the work [4] that decomposes the video description into three hierarchical semantic levels, capturing global events, local actions and entities respectively. Specifically, given a video description S that includes m words $\{s_1, \cdots, s_m\}$, we first employ an bidirectional GRU [5] (actually used by [4]) to generate the corresponding contextual-aware word embeddings $\{w_1, ..., w_m\}$. The first level of the hierarchical text graph is the global sentence representation of S, which captures event level information of the video description. We utilize an attention mechanism to aggregate the word embeddings to focus on important events in the sentence, thus, the global sentence representation S_g is given by:

$$\begin{aligned} S_g &= \sum_{i=1}^{m} \alpha_{g,i} w_i \\ \alpha_{g,i} &= \frac{\exp\left(W_g w_i\right)}{\sum_{j=1}^{m} \exp\left(W_g w_j\right)} \end{aligned} \tag{1}$$

where W_g is the learnable weight parameter. In order to obtain more fine-grained semantic information, a semantic role parsing toolkit [36] is utilized to extract verbs, noun phrases and their semantic role relations in the sentence, for instance, <girl>-<puts back>-<product>. As events are composed of different actions, the second level of the text graph is action level, whose nodes are verb phrases. Then the remaining third level is naturally the entity level, whose nodes are noun phrases. For action and entity nodes, we apply max pooling over words in each node as action node representations $S_a = \{s_{a,1}, ..., s_{a,n_a}\}$ and entity node representations $S_e = \{s_{e,1}, ..., s_{e,n_e}\}$, where n_a and n_e are numbers of action and entity nodes respectively. As for the edge connections, the action nodes are connected to the sentence node with direct edges. Since the sentence node includes global information, the contextual relations between the action nodes can be implicitly learned from sentence node during the graph reasoning. And the edge between entity node and action node is decided by the semantic role of the entity in reference to the action. In Figure 2, we give an example of the constructed hierarchical text graph.

After constructing the text graph, a graph attention network is used to learning the semantic interactions between the nodes of different levels. Given the initialized node representations $s_i \in \{S_g, S_a, S_e\}$, the graph attention network is used to select contextual information from neighbor nodes to enhance the representation for each node:

$$\tilde{\alpha}_{ij} = \left(W_a^1 s_i^l\right)^T \left(W_a^2 s_j^l\right) / \sqrt{D} \tag{2}$$

$$\alpha_{ij} = \frac{\exp\left(\tilde{\alpha}_{ij}\right)}{\sum_{k \in \mathcal{N}_i} \exp\left(\tilde{\alpha}_{ik}\right)} \tag{3}$$

where N_i is indicates of neighborhood nodes of node i, W_a^1 and W_a^2 are weight parameters of the graph attention, s_i^l is the output representation of node i at l-th graph reasoning layer, and D is the dimension of the node representation. Then, a shared transformation matrix W_t is utilized to propagate contextual information from attended neighbor nodes to node i with a residual connection:

$$s_i^{l+1} = s_i^l + W_t \sum_{j \in \mathcal{N}_i} \left(\alpha_{ij} s_j^l\right) \tag{4}$$

After the attention-based graph reasoning process, we can obtain the final node representations of the hierarchical text graph, which are used for our cross-modal graph consistency learning.

3.2 Hierarchical Video Graph

Different from the text, parsing video directly into a hierarchical structure can be challenging. The previous work [4] use three independent linear transformations to encode video frames into three levels of representations. Although such operation is straightforward, it may not take full advantage of the hierarchical text graph. Thus, we propose a truly hierarchical video graph representation, which also includes three levels: global video level, clip level and object level, corresponding to the three levels of the text graph.

Specifically, given an input video V as a sequence of frames, we first extract the frame features, denoted by $F = \{f_1, \cdots, f_n\}$. For the global video level, we utilize an attention-aware aggregation mechanism to aggregate significant frame features into one global vector representation V_g, similar to Equation 1. And the second level of our hierarchical video graph consists of video clips, which contain a single camera shot or a continuous action segment. In order to obtain these video clips, we use a temporal segmentation algorithm [34] by replacing the similarity matrix with new frame feature similarities. It utilizes dynamic programming to minimize within-segment kernel variances, the object function is given by:

$$\underset{u;t_0,\ldots,t_{u-1}}{\text{Minimize}} \quad J_{u,n} := L_{u,n} + \beta g(u,n) \tag{5}$$

where u is the number of change points of clips, t_i is frame index (t_{-1}, t_u are begin and end index), n is number of frames, β is a parameter and $g(u,n) = u(\log(n/u) + 1)$ is a penalty term. $L_{u,n}$ is defined from the within-segment kernel variances v_{t_{i-1},t_i}:

$$L_{u,n} = \sum_{i=0}^{u} v_{t_{i-1},t_i}, \tag{6}$$

$$v_{t_{i-1},t_i} = \sum_{j=t_{i-1}}^{t_i-1} \left\| f_j - \gamma \right\|^2, \quad \gamma = \frac{\sum_{k=t_{i-1}}^{t_i-1} f_k}{t_i - t_{i-1}} \tag{7}$$

The obtained video clips are denoted as $c_\tau = \{f_{\tau,1}, f_{\tau,2}, \ldots, f_{\tau,n_\tau}\}$, where $\tau = 1, 2, \ldots, Y$, Y is the number of clips and n_τ is the number of frames of clip τ. Then, we utilize the clip aggregation mechanism to aggregate significant frame features into clip representations $V_c = \{v_{c,1}, \ldots, v_{c,Y}\}$, similar to Equation 1. Besides, we use an average pooling over the frames of clips to get the pooling representations, denoted as $V_p = \{v_{p,1}, \ldots, v_{p,Y}\}$, and more complex pooling operation can aslo be used for better representations. Due to the potential semantic relation among the clips, we build a clip graph to learn such information. Specifically, we utilize the clip pooling features V_p to calculate similarities of the adjacency matrix A:

$$A_{i,j} = \text{Softmax}((W_s^1 v_{p,i})^T (W_s^2 v_{p,j})) \tag{8}$$

where W_s^1 and W_s^2 are learnable weights. After that, we use a graph convolutional network (GCN) to update the clip representations in the clip graph, given by:

$$V_c^{l+1} = \sigma\left(\tilde{D}^{-\frac{1}{2}} A \tilde{D}^{-\frac{1}{2}} V_c^l W_c\right) \tag{9}$$

where $\tilde{D}$ is the diagonal degree matrix of A and W_c is the weight matrix. As for the object level, we first extract a number of object region proposals from each sampled frame using an off-the-shelf object detector, denoted as $R_t = \{r_{t,1}, r_{t,2}, \ldots, r_{t,n_t}\}$, where $t = 1, 2, \ldots, n$, n is the number of sampled frames and n_t is the number of object regions of frame t. The corresponding region features are denoted as $O_t = \{o_{t,1}, o_{t,2}, \ldots, o_{t,n_t}\}$. For each frame, we employ an object aggregation mechanism to select significant object information, similar to Equation 1. Noted that more complex spatial-temporal modeling can be used in this step for extracting object interactions, which also requires more computational overhead. Then, we group the aggregated object features in each clip to obtain the object level representation, as shown in Figure 2, denoted as $V_o^\tau = \{v_{o,1}^\tau, v_{o,2}^\tau, \ldots, v_{o,n_\tau}^\tau\}$. By connecting the global video node, clip node and its object nodes group, we could obtain a hierarchical video graph representation for fine-grained cross-modal matching.

3.3 Cross-modal Graph Consistency

As shown in Figure 2, our cross-modal graph consistency matching includes three types: inter-graph parallel consistency, inter-graph cross consistency and intra-graph cross consistency.

Inter-graph parallel consistency is similar to the matching scheme introduced by [4], which aims to match the information between the corresponding levels of the video and text graph. For the global-to-global level, we use cosine similarity to measure the similarity for global video and text representations. The matching score is denoted as $Sim_g = cos(V_g, S_g)$. As for the action-to-clip (a_c) and entity-to-object (e_o) levels, there are multiple nodes in both layers. Taking action-to-clip (a_c) as an example, suppose each node $s_{a,i} \in S_a, v_{c,j} \in V_c$, we first calculate the similarities between each pair of cross-modal nodes $Sim_{ij}^{a\_c} = cos(v_{c,j}, s_{a,i})$. Then, we normalize the $Sim_{ij}^{a\_c}$ as follows:

$$\varphi_{ij}^{a\_c} = \text{softmax}(\lambda([Sim_{ij}^{a\_c}]_+ / \sqrt{\sum_j [Sim_{ij}^{a\_c}]_+^2})) \tag{10}$$

where $[\cdot]_+ \equiv \max(\cdot, 0)$. The normalized weight $\varphi_{ij}^{a\_c}$ is then used as attention weight over all clip nodes V_c for each action node $s_{a,i}$ to obtian an aggregated similarity $Sim_{i,c} = \sum_j \varphi_{ij}^{a\_c} Sim_{ij}^{a\_c}$. The final matching similarity of action-to-clip level is calculated by summarizing all aggregated similarities, denoted as $Sim_{ac} = \sum_i Sim_{i,c}$. Similarly, we can obtain the matching similarity of entity-to-object level, given by $Sim_{eo} = \sum_i Sim_{i,o}$. By averaging the above three parallel levels of cross-modal similarities, we can obtain the inter-graph parallel similarity:

$$S_{p-inter}(V, S) = \left(Sim_g + Sim_{ac} + Sim_{eo}\right)/3 \tag{11}$$

We utilize contrastive ranking loss as the training objective to learn the inter-graph parallel consistency. For each positive pair (V^+, S^+), we find the hardest negatives within a mini-batch (V^+, S^-) and (V^-, S^+), and push their distances from the positive pair (V^+, S^+) further away than a pre-defined margin Δ as follows:

$$\begin{aligned} L_p = &\left[\Delta + S_{p-inter}\left(V^+, S^-\right) - S_{p-inter}\left(V^+, S^+\right)\right]_+ \\ &+ \left[\Delta + S_{p-inter}\left(V^-, S^+\right) - S_{p-inter}\left(V^+, S^+\right)\right]_+ \end{aligned} \tag{12}$$

As a basic matching scheme, the inter-graph parallel consistency only considers the information matching between the same layer of the video and text graph. And it may not be a good choice to directly utilize a similar similarity described above to learn the consistency across different layers, since we hope different levels of graphs to provide different information for fine-grained matching. Thus, we propose another two types of graph consistency to provide indirect cross-level consistency constraints.

Inter-graph cross consistency aims to learn indirect consistency between different layers of the video and text graph. Specifically, taking the global_v-to-action as an example, $\{V_1, \ldots, V_N\}$ is a mini-batch set of videos and $\{S_1, \ldots, S_N\}$ is the corresponding description set, where N is the mini-batch size. The global level video representations are $Vg = \{V_g^1, \ldots, V_g^N\}$ and $Sa = \{S_a^1, \ldots, S_a^N\}$, $S_a^N = \{s_{a,1}^N, ..., s_{a,n_a}^N\}$ are action level representations of the texts. Given the i-th input sample, the first step is similar to the parallel consistency learning, we calculate the similarities between the V_g^i and S_a. Then, we normalize the similarities as follows:

$$\alpha_{pq} = \frac{\exp\left(\text{Sim}\left(V_g^i, s_{a,q}^p\right)\right)}{\sum_{pq}^{N \times n_a} \exp\left(\text{Sim}\left(V_g^i, s_{a,q}^p\right)\right)} \tag{13}$$

We could obtain an aggregated action representation, denoted as:

$$\widetilde{S}_a^i = \sum_{pq}^{N \times n_a} \alpha_{pq} s_{a,q}^p \tag{14}$$

Then, we regard $\widetilde{S}_a^i$ as an agent of V_g^i, thus, the consistency between the two levels becomes indirect. So, $(\widetilde{S}_a^i, s_{a,j}^i)$ can be seen as a positive pair. Our aim is to push the negative pairs $(\widetilde{S}_a^i, s_{a,q}^p)$ where $s_{a,q}^p \in Sa, p \neq i$ away from the positive pairs, which is given by:

$$L_{g2a} = \sum_{j, s_{a,j}^i \in S_a^i} -\log \frac{\exp\left(\text{Sim}\left(\widetilde{S}_a^i, s_{a,j}^i\right)/\eta\right)}{\sum_{pq}^{N \times n_a} \exp\left(\text{Sim}\left(\widetilde{S}_a^i, s_{a,q}^p\right)/\eta\right)} \tag{15}$$

The above consistency can be bidirectional. For action-to-global_v, we utilize a new aggregated (Eq 1) action $\overline{S}_a^i$ to compute the similarities with $V_g^j, j \in N$, and the corresponding loss is denoted as L_{a2g}. Thus, the complete loss becomes $L_{ga} = L_{g2a} + L_{a2g}$. Similarly, we can obtain other three losses: L_{ge}, L_{gc} and L_{go}. The inter-graph cross consistency loss is $L_{c-inter} = L_{ga} + L_{ge} + L_{gc} + L_{go}$.

Intra-graph cross consistency aims to learn indirect consistency between different layers inside the video and text graph. Here we omit the relevant calculation process, since it's almost the same as inter-graph cross consistency besides the corresponding levels. The intra-graph cross consistency loss is $L_{c-intra} = L_{ga} + L_{ge} + L_{gc} + L_{go}$.

Thus, the final loss for training is $L_{final} = L_p + \mu(L_{c-inter} + L_{c-intra})$, where μ is a parameter. And during inference, we keep same with [4], only using the direct $S_{p-inter}(V, S)$ as final video-text similarity, which can also reduce the amount of calculation.

4 EXPERIMENTS

4.1 Experimental Settings

Datasets. In our experiment, we use MSR-VTT [45], TGIF [24] and Youtube2Text [15] datasets. There are 10000 video clips from YouTube in the MSR-VTT dataset, with 20 descriptive sentences for each video, which results in 200000 unique video-text pairs available in this dataset. According to the official split, we use 6513 videos as training data, 497 videos as validation data, and 2990 videos as test data. TGIF dataset is composed of GIF format videos. There are 101412 videos in this dataset, and each video corresponds to 1 to 3 descriptive sentences. In the official standard method, there are 79451 videos of training data, 10651 videos of validation data and 11310 videos of test data. As for Youtube2Text (same as MSVD [3]) dataset, there are 1970 YouTube video clips, each corresponding to about 40 sentences. In the official spliting method, 1200 videos are use as training data, 100 videos as verification data, and 670 videos as test data. We use the test set of the Youtube2Text dataset to verify the generalization ability of our model pre-trained on the MSR-VTT dataset.

Implementation Details. We first extract the frames of the video, and each video will be extracted up to 20 frames at equal intervals. For the video feature encoding, it includes two parts: the first part we use Resnet152 pre-trained on Imagenet [18] to extract frame-wise features, the second part we use Faster-RCNN pre-trained on Visual Genome [1] to get the object regions (18 per frame) and corresponding object level features. The dimension of these two parts of features is 2048. As for the text encoding, we set the word embedding size to 300 and initialize it with pre-trained Glove embeddings [33]. We use two layers of graph reasoning operation for the text graph and one layer for the video graph, since more layers would not bring much improvements but need more computation. For video segmentation, we set the maximum number of segments as 10. And for both the video and text, the dimension of the joint embedding space in each level is set to 1024.

Evaluation Metrics. For both video retrieval and text retrieval, we use common metrics to measure our retrieval performance. The metrics we use include: Recall at K (R@K), Median Rank (MedR) and Mean Rank(MnR). R@K is the fraction of queries that correctly retrieve desired items in the top K of ranking list. We use K = 1, 5, 10 following the tradition. The MedR and MnR measure the median and average rank of correct items in the retrieved ranking list respectively. Additionally, we use the sum of all R@K as rsum to measure the overall retrieval performance. For R@K and rsum, higher score indicates better performance, and for MedR and MnR, lower score indicates better performance.

Training Details. Our experiment is carried out on the PyTorch framework. We set $\lambda = 4$ in the Equation 10. We train our model with the Adam optimizer and set the initial learning rate as 0.0001 with mini-batch size of 64. For the loss function, we set $\eta = 0.01$ and μ as 0.035. During training, we set the margin $\Delta = 0.2$, and train the model for 50 epochs. The epoch with the best rsum on validation set will be selected for inference. In the future, we also want to implement it on MindSpore, which is a new deep learning computing framework[1].

[1]https://www.mindspore.cn/

4.2 Experimental Results and Analysis

4.2.1 Comparison and Analysis. Table 1 shows the comparison results of our model and recent approaches on the MSR-VTT testing set. Specifically, ViSERN [11] pays more attention to video modeling by using a random walk rule-based graph convolutional network to generate region features involved with semantic relations. TCE [47] focuses on complex text queries by learning the linguistic structure of queries and the temporal representation of videos. And HGR [4] is the previous best reported method, which achieves fine-grained video-text retrieval through disentangling text into a hierarchical semantic graph and global-to-local level matching. However, HGR doesn't fully explore the hierarchical video modeling and multi-level cross-modal matching strategies, which are considered in our model. Thus, our model outperforms all listed approaches across all evaluation metrics on the MSR-VTT dataset, as shown in Table 1. And it can be found that the margin between our model and HGR in video-to-text retrieval is larger than it in text-to-video retrieval. These results demonstrate the effectiveness of our hierarchical graph-based video representations and multi-level cross-modal graph consistency learning. We also evaluate our model on TGIF dataset to demonstrate the robustness of our model on different datasets. For a fair comparison, we utilize the same video feature extraction network for different datasets. As shown in Table 2, we can see that the same methods have lower metrics than those in Table 1, which means TGIF dataset is more difficult than MSR-VTT. Our HCGC model not only outperforms all listed methods again, but also achieves a larger performance improvement over HGR on both video-to-text and text-to-video retrieval (gain of Rsum, 21.1 on TGIF vs 15.3 on MSR-VTT). And as a retrieval model, it's also important to be generalizable to out-of-domain data. Therefore, we first pre-train the model on one dataset and then evaluate its performance on another dataset unseen during the training. Specifically, we still use the MSR-VTT dataset as the training set but test models on the Youtube2Text testing split [15], which includes 670 videos and 41.5 descriptions per video on average. Table 3 presents the generalization experimental results. As a result, our HCGC model achieves consistent improvements across different datasets (both in-domain and out-of-domain) compared with previous models, which demonstrates the effectiveness our hierarchical video-text graph representations and multi-level cross-modal graph consistency learning for fine-grained video-text retrieval.

4.2.2 Ablation Studies. In this section, we conduct a series of ablation studies on MSR-VTT dataset to investigate contributions of different components of our model. The detailed results are shown in Table 4. We first investigate the influence of different types of matching loss independently. As a basic loss, we could find that only using inter-graph parallel consistency loss is powerful enough to outperform the HGR model, which indicates the effectiveness of our hierarchical video graph representation. The simple direct similarity cross loss has little improvements based on basic loss, as we discussed in Sec 3.3. And both intra-graph cross consistency and inter-graph cross consistency loss can help the model make further improvements based on the basic loss, because they could provide additional indirect matching information from different perspectives. Then, we remove the graph processing part of video

Table 1: Experimental results of comparison with other models on MSR-VTT dataset.

Model	Text-to-Video Retrieval					Video-to-Text Retrieval					rsum
	R@1	R@5	R@10	MedR	MnR	R@1	R@5	R@10	MedR	MnR	
VSE [20]	5.0	16.4	24.6	47	215.1	7.7	20.3	31.2	28	185.8	105.2
VSE++ [9]	5.7	17.1	24.8	65	300.8	10.2	25.4	35.1	25	228.1	118.3
JEMC [31]	5.8	17.6	25.2	61	296.6	10.5	26.7	35.9	25	266.6	121.7
W2VV [7]	6.1	18.7	27.5	45	-	11.8	28.9	39.1	21	-	132.1
Dual Encoding [8]	7.7	22.0	31.8	32	-	13.0	30.8	43.3	15	-	148.6
TCE [47]	7.7	22.5	32.1	30	-	-	-	-	-	-	-
ViSERN [11]	7.9	23.0	32.6	30	178.7	13.1	30.1	43.5	15	119.1	151.1
HGR [4]	9.2	26.2	36.5	24	164.0	15.0	36.7	48.8	11	90.4	172.4
Ours	**9.7**	**28.0**	**39.2**	**19**	**129.5**	**17.1**	**40.5**	**53.2**	**9**	**58.2**	**187.7**

Table 2: Experimental results of comparison with other models on TGIF dataset.

Model	Text-to-Video Retrieval					Video-to-Text Retrieval					rsum
	R@1	R@5	R@10	MedR	MnR	R@1	R@5	R@10	MedR	MnR	
DeViSE [12]	0.8	3.4	6.0	378	-	0.8	3.5	6.0	379	-	20.6
VSE++ [9]	0.6	1.9	3.8	620	-	0.4	1.6	3.6	692	-	11.9
Order [42]	0.5	2.1	3.9	478	-	0.5	2.1	3.8	500	-	12.9
Corr-AE [10]	0.9	3.5	6.0	352	-	0.9	3.4	5.6	365	-	20.3
PVSE [38]	2.2	7.8	12.3	155	-	2.3	7.5	11.9	162	-	43.9
HGR [4]	4.5	12.4	17.8	160	653.4	6.7	16.8	23.4	78	545.7	83.3
Ours	**6.3**	**16.2**	**22.9**	**79**	**495.7**	**9.0**	**21.3**	**28.7**	**48**	**364.5**	**104.4**

Table 3: Generalization on unseen Youtube2Text testing set using models pre-trained on MSR-VTT dataset.

Model	Text-to-Video Retrieval					Video-to-Text Retrieval					rsum
	R@1	R@5	R@10	MedR	MnR	R@1	R@5	R@10	MedR	MnR	
VSE [20]	11.0	28.6	39.9	18	48.7	15.4	31.0	42.4	19	128.0	168.3
VSE++ [9]	13.8	34.6	46.1	13	48.4	20.8	37.6	47.8	12	108.3	200.6
Dual Encoding [8]	12.7	32.0	43.8	15	52.7	18.7	37.2	45.7	15	142.6	190.0
HGR [4]	16.4	38.3	49.8	11	49.2	23.0	42.2	53.4	8	77.8	223.2
Ours	**17.4**	**39.6**	**52.6**	**9**	**42.1**	**24.2**	**47.9**	**56.4**	**6**	**77.4**	**238.1**

Table 4: Ablation studies on MSR-VTT dataset to investigate effects of different components.

Variants	Text-to-Video Retrieval					Video-to-Text Retrieval					rsum
	R@1	R@5	R@10	MedR	MnR	R@1	R@5	R@10	MedR	MnR	
inter-p loss	9.3	27.1	38.2	20	143.8	15.2	38.5	50.9	10	65.0	179.2
+simple-c loss	9.2	27.3	38.3	20	140.2	15.2	38.7	51.2	10	65.2	179.9
+intra-c loss	9.6	27.7	38.8	20	136.1	15.8	39.3	52.3	9	60.8	183.5
+inter-c loss	9.5	27.7	38.7	19	132.7	16.2	39.7	52.6	9	64.8	184.4
w/o clip-graph	9.5	27.9	39.0	19	139.0	15.6	39.0	52.6	9	64.0	183.6
object-Avg	9.4	27.5	38.4	20	132.4	14.6	38.1	51.2	10	64.5	179.2
full model	**9.7**	**28.0**	**39.2**	**19**	**129.5**	**17.1**	**40.5**	**53.2**	**9**	**58.2**	**187.7**

clips, which means we directly utilize the clip features after clip aggregation without considering the context relationships among clips. As shown in the table, this will cause performance degradation, especially on video-to-text retrieval task, which demonstrates the importance of contextual clip information. We also attempt to replace the object aggregation mechanism with a simple average pooling operation for all objects in each frame. According to the result, this will lead to more performance degradation on both video-to-text and text-to-video retrieval, because object-level information is more fine-grained and average pooling cannot adequately capture such information.

In Table 6, we break down the performance at each single level of video-text matching. We can find that the global-global level gets the best single level performance since it includes the most

Table 5: Performance of different models on fine-grained binary selection task.

Model	switch roles	replace actions	replace persons	replace scenes	incomplete events	average
# of triplets	616	646	670	539	646	623.4
VSE++ [9]	64.61	74.46	85.67	83.30	78.79	77.37
Dual Encoding [8]	71.92	71.52	86.12	82.00	70.59	76.43
HGR [4]	69.48	71.21	86.27	84.05	82.04	78.61
Ours	73.21	75.70	87.61	84.41	82.66	**80.72**

Table 6: Break down of retrieval performance at different levels on MSR-VTT testing set.

	Text-to-Video			Video-to-Text		
	rsum	MedR	MnR	rsum	MedR	MnR
event [4]	57.6	43	267.8	77.8	20.5	258.0
action [4]	50.4	77	441.6	80.7	22	241.4
entity [4]	44.7	62	251.3	58.4	37	230.0
global-global	70.8	23	168.9	98.5	11.5	104.9
action-clip	53.3	44	280.5	81.3	19	151.3
entity-object	63.3	29	176.1	90.6	14	95.0
full model	**76.9**	**19**	**129.5**	**110.8**	**9**	**58.2**

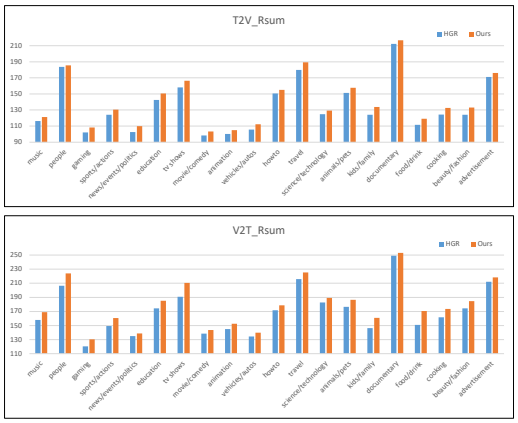

Figure 3: Performance comparison on grouped MSR-VTT testing set in terms of text categories.

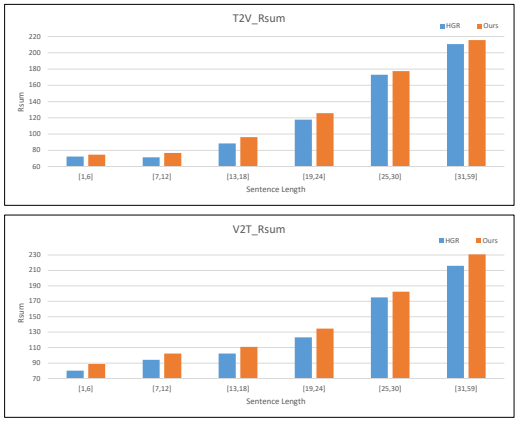

Figure 4: Performance comparison on grouped MSR-VTT testing set in terms of text lengths.

complete modality information. Although the other two levels perform a bit lower, they could provide complementary information for the global level since their combination (full model) achieves the best performance. Compared with the HGR [4] model, our approach performs better at each level, which not only reflects the effectiveness of each level, but also shows the impacts of our video representations and graph consistency on each level.

4.2.3 Qualitative Analysis. In Figure 5, we visualize three examples on the MSR-VTT testing split for text-to-video retrieval. As shown in the first example, the top 3 retrieved videos are all related to the skiing mentioned in the text. The top 1 video exactly match the description, while the second one may focus more on photo taking not skiing and the third video is total about a reporter speaking. In the second example, our model could understand the actions and their relation in the description, and also accurately distinguish the video contents. The last example shows a fail case, where the text is quite short making it difficult to distinguish similar videos (the top-2 are not really assembling parts). In Figure 6, we also provide visualization examples on video-to-text retrieval. These examples demonstrate the effectiveness of our model for video-text retrieval on both directions.

4.2.4 Other Analysis. We also evaluate our model on a binary selection task proposed by [4]. It's designed for testing fine-grained retrieval ability, which requires the model to select a sentence from two very similar but semantically different sentences according to a given video. Besides the positive sentence, the negative sentence is generated by disturbing the ground-truth sentence in several ways: switch roles (switching agent and patient of an action), replace actions (replacing action with random action), replace persons (replacing agent or patient entities with random agents or patients), replace scenes (randomly replacing scene entities), incomplete events (only keeping part of all actions, entities in the sentence). The detailed results are shown in Table 5. For the average score, our model outperforms HGR model with absolute 2.11%. Specifically, our model gets large improvements over HGR model in the switch roles and replace actions task (3.73% and 4.49%, respectively). Since our text graph module is similar to HGR model, the improvements are mainly contributed by our hierarchical video graph module and graph consistency learning strategies, which help to learn better hierarchical fine-grained modality representations.

And following the work [47], we also show the performance comparison on grouped MSR-VTT testing set according to the sentence

Figure 5: Text-to-video retrieval examples on MSR-VTT testing set with top 3 retrieved videos (green: correct; red: incorrect).

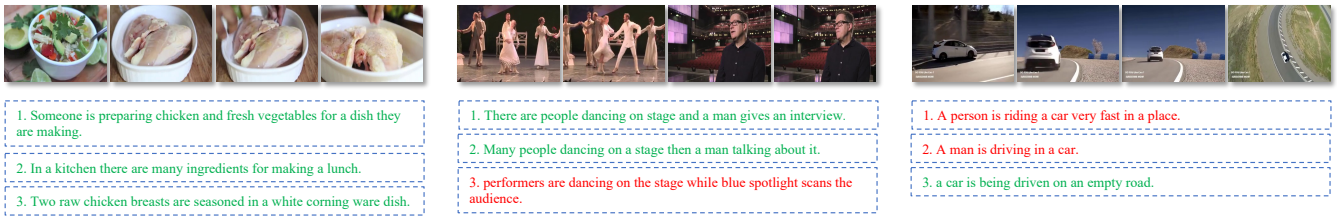

Figure 6: Video-to-text retrieval examples on MSR-VTT testing set with top 3 retrieved texts (green: correct; red: incorrect).

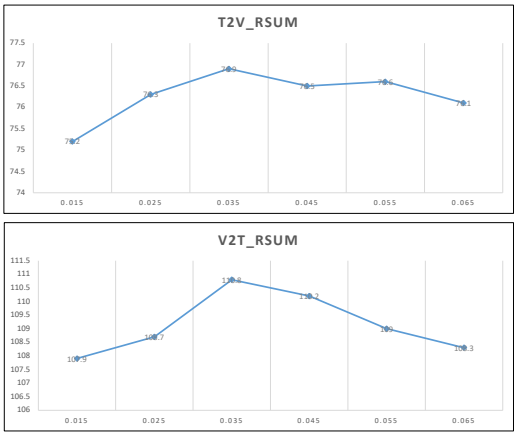

Figure 7: The influence of different values of μ on the 'Rsum' for both text-to-video and video-to-text retrieval.

lengths and categories. As Figure 3 shows, the tendency of results in text-to-video and video-to-text retrieval task are quite similar, our HCGC consistently beats the HGR on all categories. One difference is that the performance gain in video-to-text retrieval is larger than in text-to-video retrieval, which indicates the effectiveness of our video modeling. As shown in Figure 4, our model consistently outperforms the HGR in all groups of different sentence lengths. Especially on video-to-text retrieval task, as the sentence length increases, the performance gap becomes larger, showing the effectiveness of our model in complex semantic information retrieval. And we can find that although the longer sentence is generally more complicated, the retrieval performance of the short sentence is lower instead. One possible reason we think is that the short sentence contains less effective information to distinguish similar video contents, which is not conducive to accurate retrieval. In Figure 7, we also show the influence of different values of μ on the 'Rsum' for both text-to-video and video-to-text retrieval. According to the results, we find $\mu = 0.035$ is a relatively appropriate value.

5 CONCLUSION

In this work, we propose a hierarchical cross-modal graph consistency learning network (HCGC) for video-text retrieval task, which considers multi-level graph consistency for video-text matching. Specifically, we first construct a hierarchical graph representation for the video, which includes three levels from global to local: video, clips and objects. Similarly, the corresponding text graph is constructed according to the semantic relationships among sentence, actions and entities level. Then, in order to learn a better match between the video and the text graph, we design three types of graph consistency (both direct and indirect): inter-graph parallel consistency, inter-graph cross consistency and intra-graph cross consistency. Extensive experimental results on different video-text retrieval datasets demonstrate the effectiveness of our approach on both text-to-video and video-to-text retrieval. In the future, we will improve our approach by exploring more modalities in the video (such as audio) and more complex cross-modal graph structure.

ACKNOWLEDGMENTS

This work was supported in part by the National Key R&D Program of China under Grant No.2020YFC0832505, National Natural Science Foundation of China under Grant No.61836002, No.62072397 and Zhejiang Natural Science Foundation under Grant LR19F020006.

REFERENCES

[1] Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. 2018. 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*. 6077–6086.

[2] Xiaojun Chang, Yi Yang, Alexander Hauptmann, Eric P Xing, and Yao-Liang Yu. 2015. Semantic concept discovery for large-scale zero-shot event detection. In *Twenty-fourth international joint conference on artificial intelligence*.

[3] David Chen and William B Dolan. 2011. Collecting highly parallel data for paraphrase evaluation. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*. 190–200.

[4] Shizhe Chen, Yida Zhao, Qin Jin, and Qi Wu. 2020. Fine-grained Video-Text Retrieval with Hierarchical Graph Reasoning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 10638–10647.

[5] Kyunghyun Cho, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 1724–1734.

[6] Jeffrey Dalton, James Allan, and Pranav Mirajkar. 2013. Zero-shot video retrieval using content and concepts. In *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*. 1857–1860.

[7] Jianfeng Dong, Xirong Li, and Cees GM Snoek. 2018. Predicting visual features from text for image and video caption retrieval. *IEEE Transactions on Multimedia* 20, 12 (2018), 3377–3388.

[8] Jianfeng Dong, Xirong Li, Chaoxi Xu, Shouling Ji, Yuan He, Gang Yang, and Xun Wang. 2019. Dual encoding for zero-example video retrieval. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 9346–9355.

[9] Fartash Faghri, David J Fleet, Jamie Ryan Kiros, and Sanja Fidler. 2017. Vse++: Improving visual-semantic embeddings with hard negatives. *arXiv preprint arXiv:1707.05612* (2017).

[10] Fangxiang Feng, Xiaojie Wang, and Ruifan Li. 2014. Cross-modal retrieval with correspondence autoencoder. In *Proceedings of the 22nd ACM international conference on Multimedia*. 7–16.

[11] Zerun Feng, Zhimin Zeng, Caili Guo, and Zheng Li. 2020. Exploiting Visual Semantic Reasoning for Video-Text Retrieval. In *Proceedings of the International Joint Conference on Artificial Intelligence*.

[12] Andrea Frome, Greg S Corrado, Jon Shlens, Samy Bengio, Jeff Dean, Marc'Aurelio Ranzato, and Tomas Mikolov. 2013. Devise: A deep visual-semantic embedding model. In *Advances in Neural Information Processing Systems*. 2121–2129.

[13] Yunchao Gong, Qifa Ke, Michael Isard, and Svetlana Lazebnik. 2014. A multi-view embedding space for modeling internet images, tags, and their semantics. *International journal of computer vision* 106, 2 (2014), 210–233.

[14] Jiuxiang Gu, Jianfei Cai, Shafiq R Joty, Li Niu, and Gang Wang. 2018. Look, imagine and match: Improving textual-visual cross-modal retrieval with generative models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 7181–7189.

[15] Sergio Guadarrama, Niveda Krishnamoorthy, Girish Malkarnenkar, Subhashini Venugopalan, Raymond Mooney, Trevor Darrell, and Kate Saenko. 2013. Youtube2text: Recognizing and describing arbitrary activities using semantic hierarchies and zero-shot recognition. In *Proceedings of the IEEE international conference on computer vision*. 2712–2719.

[16] Amirhossein Habibian, Thomas Mensink, and Cees GM Snoek. 2014. Composite concept discovery for zero-shot video event detection. In *Proceedings of International Conference on Multimedia Retrieval*. 17–24.

[17] Amirhossein Habibian, Thomas Mensink, and Cees GM Snoek. 2016. Video2vec embeddings recognize events when examples are scarce. *IEEE transactions on pattern analysis and machine intelligence* 39, 10 (2016), 2089–2103.

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

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

[20] Ryan Kiros, Ruslan Salakhutdinov, and Richard S Zemel. 2014. Unifying visual-semantic embeddings with multimodal neural language models. *arXiv preprint arXiv:1411.2539* (2014).

[21] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2012. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems* (2012).

[22] Duy-Dinh Le, Sang Phan, Vinh-Tiep Nguyen, Benjamin Renoust, Tuan A Nguyen, Van-Nam Hoang, Thanh Duc Ngo, Minh-Triet Tran, Yuki Watanabe, Martin Klinkigt, et al. 2016. NII-HITACHI-UIT at TRECVID 2016.. In *TRECVID*.

[23] Xirong Li, Chaoxi Xu, Gang Yang, Zhineng Chen, and Jianfeng Dong. 2019. W2vv++ fully deep learning for ad-hoc video search. In *Proceedings of the 27th ACM International Conference on Multimedia*. 1786–1794.

[24] Yuncheng Li, Yale Song, Liangliang Cao, Joel Tetreault, Larry Goldberg, Alejandro Jaimes, and Jiebo Luo. 2016. TGIF: A new dataset and benchmark on animated GIF description. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 4641–4650.

[25] Dahua Lin, Sanja Fidler, Chen Kong, and Raquel Urtasun. 2014. Visual semantic search: Retrieving videos via complex textual queries. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2657–2664.

[26] Chunxiao Liu, Zhendong Mao, Tianzhu Zhang, Hongtao Xie, Bin Wang, and Yongdong Zhang. 2020. Graph Structured Network for Image-Text Matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 10921–10930.

[27] Yang Liu, Samuel Albanie, Arsha Nagrani, and Andrew Zisserman. 2019. Use what you have: Video retrieval using representations from collaborative experts. *arXiv preprint arXiv:1907.13487* (2019).

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

[29] Foteini Markatopoulou, Damianos Galanopoulos, Vasileios Mezaris, and Ioannis Patras. 2017. Query and keyframe representations for ad-hoc video search. In *Proceedings of the 2017 ACM on International Conference on Multimedia Retrieval*. 407–411.

[30] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781* (2013).

[31] Niluthpol Chowdhury Mithun, Juncheng Li, Florian Metze, and Amit K Roy-Chowdhury. 2018. Learning joint embedding with multimodal cues for cross-modal video-text retrieval. In *Proceedings of the 2018 ACM on International Conference on Multimedia Retrieval*. 19–27.

[32] Yingwei Pan, Tao Mei, Ting Yao, Houqiang Li, and Yong Rui. 2016. Jointly modeling embedding and translation to bridge video and language. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 4594–4602.

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

[34] Danila Potapov, Matthijs Douze, Zaid Harchaoui, and Cordelia Schmid. 2014. Category-specific video summarization. In *European conference on computer vision*. 540–555.

[35] Thi Quynh Nhi Tran, Hervé Le Borgne, and Michel Crucianu. 2016. Aggregating image and text quantized correlated components. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2046–2054.

[36] Peng Shi and Jimmy Lin. 2019. Simple bert models for relation extraction and semantic role labeling. *arXiv preprint arXiv:1904.05255* (2019).

[37] Cees GM Snoek and Marcel Worring. 2009. Concept-Based Video Retrieval. *Foundations and Trends in Information Retrieval* 2, 4 (2009), 215–322.

[38] Yale Song and Mohammad Soleymani. 2019. Polysemous visual-semantic embedding for cross-modal retrieval. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 1979–1988.

[39] Weijie Su, Xizhou Zhu, Yue Cao, Bin Li, Lewei Lu, Furu Wei, and Jifeng Dai. 2020. VL-BERT: Pre-training of Generic Visual-Linguistic Representations. In *International Conference on Learning Representations*.

[40] Yudong Tao, Tianyi Wang, Diana Machado, Raul Garcia, Yuexuan Tu, Maria Presa Reyes, Yeda Chen, Haiman Tian, Mei-Ling Shyu, and Shu-Ching Chen. 2019. Florida International University-University of Miami TRECVID 2019. *TRECVID. NIST, USA* (2019).

[41] Kazuya Ueki, Koji Hirakawa, Kotaro Kikuchi, Tetsuji Ogawa, and Tetsunori Kobayashi. 2017. Waseda_Meisei at TRECVID 2017: Ad-hoc Video Search.. In *TRECVID*.

[42] Ivan Vendrov, Ryan Kiros, Sanja Fidler, and Raquel Urtasun. 2015. Order-embeddings of images and language. *arXiv preprint arXiv:1511.06361* (2015).

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

[44] Hao Wu, Jiayuan Mao, Yufeng Zhang, Yuning Jiang, Lei Li, Weiwei Sun, and Wei-Ying Ma. 2019. Unified visual-semantic embeddings: Bridging vision and language with structured meaning representations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 6609–6618.

[45] Jun Xu, Tao Mei, Ting Yao, and Yong Rui. 2016. Msr-vtt: A large video description dataset for bridging video and language. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 5288–5296.

[46] Ran Xu, Caiming Xiong, Wei Chen, and Jason Corso. 2015. Jointly modeling deep video and compositional text to bridge vision and language in a unified framework. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

[47] Xun Yang, Jianfeng Dong, Yixin Cao, Xun Wang, Meng Wang, and Tat-Seng Chua. 2020. Tree-Augmented Cross-Modal Encoding for Complex-Query Video Retrieval. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*. 1339–1348.

[48] Youngjae Yu, Jongseok Kim, and Gunhee Kim. 2018. A joint sequence fusion model for video question answering and retrieval. In *Proceedings of the European*

Conference on Computer Vision (ECCV). 471–487.

[49] Jin Yuan, Zheng-Jun Zha, Yan-Tao Zheng, Meng Wang, Xiangdong Zhou, and Tat-Seng Chua. 2011. Utilizing related samples to enhance interactive concept-based video search. 13, 6 (2011), 1343–1355.

[50] Shengyu Zhang, Dong Yao, Zhou Zhao, Tat-Seng Chua, and Fei Wu. 2021. CauseRec: Counterfactual User Sequence Synthesis for Sequential Recommendation. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*.