Reasoning with Heterogeneous Graph Alignment for Video Question Answering*

Pin Jiang, Yahong Han†
College of Intelligence and Computing
Tianjin University, Tianjin, China
{jpin, yahong}@tju.edu.cn

Abstract

The dominant video question answering methods are based on fine-grained representation or model-specific attention mechanism. They usually process video and question separately, then feed the representations of different modalities into following late fusion networks. Although these methods use information of one modality to boost the other, they neglect to integrate correlations of both inter- and intra-modality in an uniform module. We propose a deep heterogeneous graph alignment network over the video shots and question words. Furthermore, we explore the network architecture from four steps: representation, fusion, alignment, and reasoning. Within our network, the inter- and intra-modality information can be aligned and interacted simultaneously over the heterogeneous graph and used for cross-modal reasoning. We evaluate our method on three benchmark datasets and conduct extensive ablation study to the effectiveness of the network architecture. Experiments show the network to be superior in quality.

1 Introduction

Video question answering (VideoQA), which aims to automatically infer the correct answer given a video and a related textual question, has received an increasing amount of attention in recent years (Jang et al. 2019). VideoQA's inference always involves heterogeneous data from two domains, i.e., spatio-temporal video content and word sequence in language.

Recent efforts towards VideoQA (Tapaswi et al. 2016; Jang et al. 2017) try to uncover latent correlations between video content and words' semantics, which could be taken as inter-modality correlations. Li et al. introduced a specific co-attention mechanism to attend to relevant video and language. Kim et al. proposed a progressive attention memory to conduct dynamic modality fusion.

Meanwhile, it has been shown that appropriately incorporating correlations inside videos or dependencies among word sequence do help improve the performance of VideoQA, which could be regarded as exploiting intra-modality correlations. A common practice is to encode video and word sequences using RNN-based encoders separately (Jang et al. 2017). A further contribution is that Fan et al. proposed heterogeneous memory to fuse visual features while devised another memory to process question.

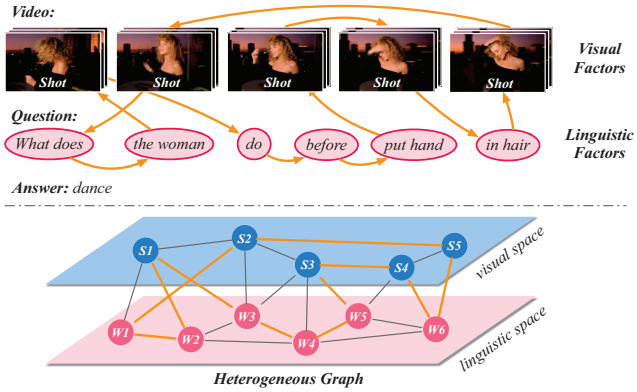

Figure 1: Given a shot-level video and a question, our model performs cross-modal reasoning over a heterogeneous graph between factors, shots s_i and words w_i, then answers it. Heterogeneous means both inter- and intra-modality. The highlighted lines indicate the reasoning process between factors.

On the other hand, in most cases, integrating correlations of both inter-modality and intra-modality (also referred to as heterogeneous relation) in a more flexible way may further benefit the inference of VideoQA, such as graph-structured methods. As shown in the upper part of Fig. 1, to answer the question, we need firstly establish semantic relations between the word *woman* and visual regions in video, then localize the action *put hand*. Moreover, we need inter-modality alignment with semantic similarity to figure out the action *dance* after temporal reasoning. However, current methods of VideoQA lack an uniform model for simultaneously modeling and reasoning with inter- and intra-modality relations.

In this paper, we propose a novel network of heterogeneous graph alignment (HGA) to perform cross-modal reasoning and VideoQA. We firstly build an uniform hetero-

*This work is supported by the NSFC (under Grant 61876130, 61932009, U1509206).

†Corresponding author.

geneous graph over different modality factors, which is an expressive and interpretable pathway. Through the heterogeneous graph, shown in the lower part of Fig. 1, there are two types of edges, intra-modality homogeneous edges and inter-modality heterogeneous edges. We can reason inside one modality, like "$s_1 \rightleftharpoons s_3$" and "$w_1 \rightleftharpoons w_2$", and inter-modality, like "$s_1 \rightleftharpoons w_2$". Over the graph, particularly, we introduce modular co-attention embedding operation to align the visual and linguistic representations, while employing further aligned Graph Convolutional Network (Kipf and Welling 2016) to model complex correlation and reason among multiple modalities. Nonetheless, an intractable drawback is the semantic gap between different modalities, blocking inter-modality interactions. Recently, cross-modal attention mechanisms are widely used as a compromise (Yu et al. 2019), and we regard the attention-based fusion as semantic alignment in an interaction space, which is crucial prior knowledge for constructing the graph. We evaluate our method on three benchmark datasets and conduct extensive ablation study to the effectiveness. Experiments show the HGA network to be superior in quality.

Our contributions can be summarized into threefold. (1) We introduce a novel heterogeneous graph to VideoQA task, representing video shots and words to graph nodes, which enables us to represent rich inter- and intra-modality interactions. (2) We propose a variety of modular co-attention embedding operations, which play an important role in cross-modal fusion and alignment. (3) We embed these modules into an overall parallel network to perform four steps: representation, fusion, alignment, and reasoning.

2 Related Work

2.1 Video Question Answering

Recently, there has been significant progress in multimodal question answering, the most representative task is Visual Question Answering (VQA) (Cadene et al. 2019; Gao et al. 2019; Yu et al. 2019) and Video Question Answering (VideoQA), where VideoQA extends VQA to video domain and raises higher demands on spatio-temporal understanding and reasoning. Tapaswi et al. adopted memory network to attend and reuse the relevant information regarding the questions. Jang et al. proposed to use spatio-temporal attention mechanism. Lei et al. introduced a multi-stream end-to-end network and used a RNN to fuse them. There are several widely used benchmark datasets. The TGIF-QA dataset is built on short, action-specific video clips and requires fine-grained action understanding and reasoning while the other two have more complicated video plots, requiring more on long-range understanding of the scenes. More recently, part of the contributions applied the dynamic memory network to enhance the intelligence through better representation and fusion strategies (Wang et al. 2018; Gao et al. 2018; Fan et al. 2019; Kim et al. 2019). Besides, Xue et al. proposed tree-structured memory network and Li et al. used self-attention to model temporal information, introducing several novel methods to VideoQA. However, existing methods focus on multimodal representation and fusion, there is still little work on alignment and reasoning.

2.2 Multimodal Information Fusion

Multimodal fusion is an original topic in multimodal machine learning. The simplest example is a vector operation of individual modality features, referred to as early fusion, including vector concatenation, element-wise addition and element-wise multiplication. Then the outputs are projected into a joint space followed by a neural network (Baltrušaitis, Ahuja, and Morency 2018). Attention mechanism has been regarded as an effective method to enhance the interaction between modalities, attending to the important terms and avoiding noise (Zhang, Cao, and Wu 2019). The co-attention mechanism is viewed as another effective solution, and previous work has suggested a variety of task-specific structures (Nguyen and Okatani 2018; Gao et al. 2019; Yu et al. 2019; Li et al. 2019). Bilinear pooling is also an effective pathway to fuse multimodal vectors by computing the outer product (Ben-Younes et al. 2019), and provides multiplicative interaction between all elements of both vectors. The effectiveness of these methods is widely proven in the VQA and VideoQA task.

3 Method

The framework of our HGA network is depicted in Fig. 2. In our method, we argue that each word and each video shot contains equally semantic information, and can be integrated in an uniform modules. Precisely, on the whole, we design a parallel architecture including global and local fusion. To jointly model visual and linguistic factors (shots or words), we first obtain the contextualized visual and linguistic representations. Note that when we talk about a "video shot", we mean a small video segment that can be processed by a 3D convolution module and produces a single motion vector. We embed visual and linguistic vectors into a common space through a modular co-attention embedding operation. In the next heterogeneous graph reasoning part, we first propose an alignment strategy to obtain weighted adjacency matrix, and then use the adjacency matrix to construct a multilayer graph convolution network for multimodal crossover reasoning.

3.1 Visual and Linguistic Contextual Representation

Video shots have richer motion expression ability than frame-level, so we use 3D ConvNets (i.e., C3D) (Tran et al. 2015) to obtain shot-level video motion features, and in order to take into account the perception of image, we use 2D ConvNets (i.e., ResNet) (He et al. 2016) as an auxiliary view. Then, the video is represented as two feature views, appearance features $\boldsymbol{F}_A = \left\{\boldsymbol{a}_i : i \leq L_v, \boldsymbol{a}_i \in \mathbb{R}^{d_a}\right\}$ and motion features $\boldsymbol{F}_M = \left\{\boldsymbol{m}_i : i \leq L_v, \boldsymbol{m}_i \in \mathbb{R}^{d_m}\right\}$, where L_v is the number of frames and d_a, d_m are the input dimensionalities of the two views. We obtain a joint visual representation by a concatenation of the two and use two fully-connected layers to project them into a common visual space. Then the video is represented by a set of vectors $\boldsymbol{F} = \left\{\boldsymbol{f}_i : i \leq L_v, \boldsymbol{f}_i \in \mathbb{R}^d\right\}$, where d is the dimension of visual features.

For question, we follow previous work (Gao et al. 2018; Jang et al. 2019; Fan et al. 2019) and represent each word

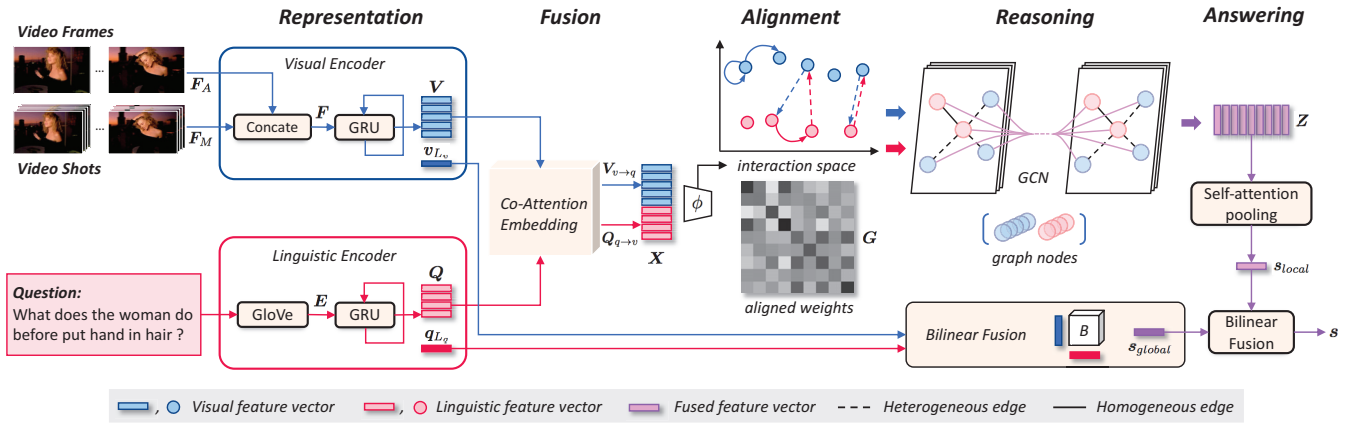

Figure 2: HGA network consists of four stages: representation, fusion, alignment, and reasoning. In representation stage, the visual encoder and linguistic encoder generate contextual representations, local $\boldsymbol{V}, \boldsymbol{Q}$ and global $\boldsymbol{v}_{L_v}, \boldsymbol{q}_{L_q}$. In fusion stage, the co-attention embedding operation performs cross-modal fusion. In alignment stage, the heterogeneous matrix $\boldsymbol{X}$ is projected to an interaction space, producing a further aligned adjacency matrix $\boldsymbol{G}$ and conducting a heterogeneous graph. In reasoning stage, the multilayer GCN takes $\boldsymbol{X}$ as inputs and performs cross-modal reasoning over $\boldsymbol{G}$ to produce local vector $\boldsymbol{s}_{local}$. In another branch, the global vector $\boldsymbol{s}_{global}$ is fed through a bilinear fusion module. Finally, they are fused together for answer.

as a vector using the pre-trained GloVe word embedding denoted as $\boldsymbol{E} = \left\{\boldsymbol{e}_i : i \leq L_q, \boldsymbol{e}_i \in \mathbb{R}^{300}\right\}$, where L_q is the number of embedded words.

In order to obtain contextual representations to aggregate dynamic temporal information from multiple time-steps and enhance reasoning ability. We employ two independent Gated Recurrent Units (GRU) to encode visual and linguistic features separately. The generated video features $\boldsymbol{V} \in \mathbb{R}^{L_v \times d}$ and question features $\boldsymbol{Q} \in \mathbb{R}^{L_q \times d}$ denote as

$$\boldsymbol{V}, \boldsymbol{v}_{L_v} = GRU\left(\boldsymbol{F}; \theta_{GRU}\right), \quad (1)$$

$$\boldsymbol{Q}, \boldsymbol{q}_{L_q} = GRU\left(\boldsymbol{E}; \theta_{GRU}\right), \quad (2)$$

where $\boldsymbol{v}_{L_v}$ and $\boldsymbol{q}_{L_q}$ are the outputs of the last hidden units which represent the global features of the two sequences.

3.2 Cross-Modal Joint Fusion and Alignment

Beyond recognizing and memorizing the visual and linguistic contents, VideoQA also requires to understand the dense interactions between factors of different modalities in a common space. We consider co-attention mechanism as one type of cross-modal fusion and alignment method between visual and textual channels, because it requires one modality as the clue to determine the weight of another modality by the semantic similarity. We first give a general describe of modular co-attention embedding operation (*CAEO*) and then we provide several universal variants of it.

Given a query and a set of key-value pairs, co-attention mechanism calculates weighted sum of values based on a compatibility function of the query and keys, and two modal features alternate as queries. Self-attention is a special case of same query and key. Supposed that query, key and value are represented as $\mathbb{M}_Q, \mathbb{M}_K, \mathbb{M}_V$, both of them are a set of vectors and packed together into matrices. Following the scaled dot-product self-attention (Vaswani et al. 2017), we define a generic *CAEO* (see Fig. 3f) as:

$$CAEO\left(\mathbb{M}_Q, \mathbb{M}_K, \mathbb{M}_V\right) = softmax\left(\frac{\mathbb{M}_Q \mathbb{M}_K^T}{\sqrt{d}}\right)\mathbb{M}_V, \quad (3)$$

where $\mathbb{M}_Q, \mathbb{M}_K$ usually indicate two different modalities and $\mathbb{M}_K$ and $\mathbb{M}_V$ are equal in most cases. *CAEO* embeds the information of the query into key's feature space by calculating the similarity between the two modalities and performing soft selection.

We argue that a reasonable explanation for the *CAEO* is the soft nearest neighbors theory (Goldberger et al. 2005). The softmax dot-product operation first scaled to have unit norm is equivalent to cosine similarity while soft KNN uses a softmax over Euclidean distances. The significance of this operation is that for each vector in the query, the weighted sum of value vectors is taken as its soft nearest neighbor. Therefore, the outputs of *CAEO* can be seen as vectors in the space of $\mathbb{M}_V$, but the information of query $\mathbb{M}_Q$ is embedded.

Next we discuss diversiform choices for *CAEO*, as illustrated in Fig. 3.

Transform from linguistic to visual space. We first use a linear projection to affine $\boldsymbol{Q}$ and $\boldsymbol{V}$ into a transformed space. Then we apply *CAEO* to transform linguistic representation $\boldsymbol{Q}$ to visual representation $\boldsymbol{Q}_{q\to v}$, named LinguisticVisual,

$$\boldsymbol{Q}_{CAEO} = CAEO\left(\boldsymbol{W}_q^q \boldsymbol{Q}, \boldsymbol{W}_k^q \boldsymbol{V}, \boldsymbol{W}_v^q \boldsymbol{V}\right), \quad (4)$$

$$\boldsymbol{Q}_{q\to v} = LayerNorm\left(FF_q\left(\boldsymbol{Q}_{CAEO}\right) + \boldsymbol{Q}\right). \quad (5)$$

In the above formulation, $\boldsymbol{W}_q, \boldsymbol{W}_k, \boldsymbol{W}_v$ with superscript are the learned weight matrices, FF_q is a feed-forward module implemented as linear transformation. *LayerNorm* is used here to stabilize training (Ba, Kiros, and Hinton 2016).

Transform from visual to linguistic space. Contrary to the previous practice, we transform visual representation $\boldsymbol{V}$

to linguistic representation $\boldsymbol{V}_{v\to q}$, named VisualLinguistic:

$$\boldsymbol{V}_{CAEO} = CAEO\left(\boldsymbol{W}_q^v\boldsymbol{V}, \boldsymbol{W}_k^v\boldsymbol{Q}, \boldsymbol{W}_v^v\boldsymbol{Q}\right), \quad (6)$$

$$\boldsymbol{V}_{v\to q} = LayerNorm\left(FF_v\left(\boldsymbol{V}_{CAEO}\right) + \boldsymbol{V}\right). \quad (7)$$

Co-attention transformation. Beyond the first two unidirectional affine transformations, we argue that the crossover transformation is crucial to fuse and align the information of different modalities. We combine the above two transformation and introduce a symmetrical co-attention operation, illustrated in Fig. 3c and named CoAttn.

Pseudo-siamese co-attention transformation. We utilize Eq. (4) and Eq. (6) for $\boldsymbol{Q}_{CAEO}$ and $\boldsymbol{V}_{CAEO}$, then devise two streams to generate co-attention outputs $\boldsymbol{Q}_{q\to v}$ and $\boldsymbol{V}_{v\to q}$:

$$\boldsymbol{Q}_{q\to v} = LayerNorm\left(FF_q\left([\boldsymbol{Q}_{CAEO}; \boldsymbol{Q}]\right) + \boldsymbol{Q}\right), \quad (8)$$

$$\boldsymbol{V}_{v\to q} = LayerNorm\left(FF_v\left([\boldsymbol{V}_{CAEO}; \boldsymbol{V}]\right) + \boldsymbol{V}\right), \quad (9)$$

where $[\cdot]$ is concatenation operation. FF_q, FF_v are two independent feed-forward networks, the structure is named PseudoSiamese (see Fig. 3d).

Siamese co-attention transformation is similar to PseudoSiamese, named Siamese and shown in Fig. 3e. The only difference is Siamese uses the weight-shared feed-forward network, FF_q and FF_v in Eq. (8) and (9).

Note that we have added residual connection (He et al. 2016) in each *CAEO* variant, so these modules can be stacked multiple layers for better performance. We write the outputs of this module as $\boldsymbol{Q}_{update}$ and $\boldsymbol{V}_{update}$ uniformly.

3.3 Heterogeneous Graph Reasoning

In this section, we introduce our key innovations that we devise a heterogeneous graph for cross-modal relational reasoning. We argue that cross-embedded visual factors (video shots) and linguistic factors (words) have coordinate semantic information and can be aligned for interactive reasoning in graph-structured neural networks.

Based on previous representation and fusion steps, we obtain two cross-embedded features of linguistic modality and visual modality, $\boldsymbol{Q}_{update}$ and $\boldsymbol{V}_{update}$. We seek to construct an undirected heterogeneous graph with each video shot and each question word as a node. We concatenate the $\boldsymbol{V}_{update}$ and $\boldsymbol{Q}_{update}$ to construct the heterogeneous input matrix $\boldsymbol{X}$ involving all the visual and linguistic vectors.

$$\boldsymbol{X} = \begin{bmatrix} \boldsymbol{Q}_{update} \\ \boldsymbol{V}_{update} \end{bmatrix}, \quad (10)$$

where $\boldsymbol{X} \in \mathbb{R}^{N\times d}$ and $N = L_v + L_q$. We treat the feature vectors $\{\boldsymbol{x}_i\}_{i=1}^N$ in $\boldsymbol{X}$ as nodes. Then, the heterogeneous graph is defined as $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ with N nodes $\boldsymbol{v}_i \in \boldsymbol{X}$ and fully-connected edges $(\boldsymbol{v}_i, \boldsymbol{v}_j)$.

Cross-modal aligned adjacency matrix. The initialization of edge weights, formalized as an adjacency matrix, is an crucial prior knowledge for graph. Because a question tends to focus on part of video rather than the whole, while visual and linguistic factors are also not equally weighted inside the respective modalities. For instance, practical words

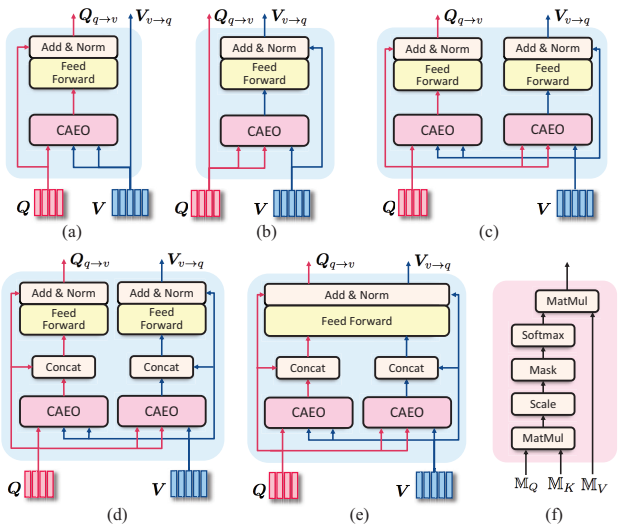

Figure 3: Flowcharts of *CAEO* variants. (a) LinguisticVisual. (b) VisualLinguistic. (c) CoAttn. (d) PseudoSiamese. (e) Siamese. (f) Vanilla *CAEO*.

are usually more important than function words. In the heterogeneous graph $\mathcal{G}$, we perform graph-based alignment to obtain cross-modal aligned adjacency matrix weighted by semantic similarity.

Firstly, we project the input features $\boldsymbol{X}$ into an interaction space by a non-linear transformation operation $\phi(\cdot)$. Then, we obtain the correlation scores of nodes by calculating the dot-product similarity (Wang and Gupta 2018):

$$\boldsymbol{G} = \phi\left(\boldsymbol{X}\right)\phi\left(\boldsymbol{X}\right)^T, \quad (11)$$

where $\boldsymbol{G}$ is defined as the adjacency matrix and $G_{i,j}$ indicates the alignment weight between $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$. Then we perform normalization on each row of the matrix by a softmax function to obtain the final aligned adjacency matrix.

By introducing the learnable transformation $\phi(\cdot)$, it is effective to learn the weights between both homogeneous and heterogeneous edges, which allows us to carry out both inter- and intra-modality alignment.

Reasoning on heterogeneous graph. Recent VQA researches point out that Graph Convolutional Network works effectively on intra-modality reasoning (Teney, Liu, and van den Hengel 2017; Norcliffe-Brown, Vafeias, and Parisot 2018). These methods usually regard the features of image regions as graph nodes and establish graph-based reasoning models between different regions. While for VideoQA, the video shots contains motion and reasoning clues, as well as words of question. Reasoning over the heterogeneous graph $\mathcal{G}$ can be natural and effective. GCN performs relational reasoning over a graph where the response of each node is updated by a linear transformation of aggregated excitation of its neighbors and itself. The weights are specified by the aligned adjacency matrix $\boldsymbol{G}$. In order to incorporate the graph input signals, we represent one layer of GCN as

$$\boldsymbol{Z} = \boldsymbol{G}\boldsymbol{X}\boldsymbol{W}, \quad (12)$$

Table 1: State-of-the-art comparison on TGIF-QA dataset. Mean ℓ_2 loss for Count, and accuracy (%) for others.

Methods	Count	Action	Trans.	FrameQA
Random	19.62	20.00	20.00	0.06
ST-VQA-Sp.	4.28	57.3	63.7	45.5
ST-VQA-Tp.	4.40	60.8	67.1	49.3
ST-VQA-Sp.Tp.	4.56	57.0	59.6	47.8
CT-SAN	5.14	56.1	64.0	39.6
Co-Mem	4.10	68.2	74.3	51.5
PSAC	4.27	70.4	76.9	**55.7**
Fan et al.	4.10[1]	73.9	77.8	53.8
ST-VQA⋆	4.22	73.5	79.7	52.0
Ours HGA	**4.09**	**75.4**	**81.0**	55.1

Table 2: State-of-the-art comparison on MSVD-QA dataset. Mean ℓ_2 loss for Count, and accuracy (%) for others.

Methods	What (8,149)	Who (4,552)	How (370)	When (58)	Where (28)	All (13,157)
ST-VQA	18.1	50.0	**83.8**	72.4	28.6	31.3
Co-Mem	19.6	48.7	81.6	**74.1**	31.7	31.7
AMU	20.6	47.5	83.5	72.4	**53.6**	32.0
Fan et al.	22.4	50.1	73.0	70.7	42.9	33.7
Ours HGA	**23.5**	**50.4**	83.0	72.4	46.4	**34.7**

Table 3: State-of-the-art comparison on MSRVTT-QA dataset. Mean ℓ_2 loss for Count, and accuracy (%) for others.

Methods	What (49,869)	Who (20,385)	How (1,640)	When (677)	Where (250)	All (72,821)
ST-VQA	24.5	41.2	78.0	**76.5**	34.9	30.9
Co-Mem	23.9	42.5	74.1	69.0	**42.9**	32.0
AMU	26.2	43.0	80.2	72.5	30.0	32.5
Fan et al.	26.5	43.6	82.4	76.0	28.6	33.0
Ours HGA	**29.2**	**45.7**	**83.5**	75.2	34.0	**35.5**

where $\boldsymbol{W} \in \mathbb{R}^{d\times d}$ is the learnable weight matrix and $\boldsymbol{Z} \in \mathbb{R}^{N\times d}$ is the output of GCN with the same shape as $\boldsymbol{X}$.

We apply a self-attention pooling on $\boldsymbol{Z}$ to obtain a local result vector $\boldsymbol{s}_{local}$ which reflects the underlying cross-modal relations after local reasoning. We define the self-attention pooling as an operation flow $\rho :=$ *(FC(d)-Tanh-FC(1)-Softmax)*, and the outputs are attended weights. So we have

$$\boldsymbol{s}_{local} = \sum^{N} \rho(\boldsymbol{Z})^T \boldsymbol{Z}. \tag{13}$$

One thing to note is that similar work (Norcliffe-Brown, Vafeias, and Parisot 2018) targeting VQA tasks also utilized question conditioned GCN to deal with the relationships between different visual regions but they did it simply by concatenating question embedding with each region's feature, while we view shots and words as equally valuable nodes.

3.4 Global and Local Information Fusion

Because the graph-structured reasoning module focuses more on the interaction between local factors, it loses the integration ability of global semantic information to some extent. In this section, we fuse the local and global information by a bilinear fusion module (Ben-Younes et al. 2019). In the first step, the last GRU hidden states, $\boldsymbol{q}_{L_q}$ and $\boldsymbol{v}_{L_v}$, are fused into a global representation:

$$\boldsymbol{s}_{global} = Bilinear\left(\boldsymbol{q}_{L_q}, \boldsymbol{v}_{L_v}\right). \tag{14}$$

Next, we use bilinear module again to fuse the global and local representations to produce the output vector $\boldsymbol{s}$ of the whole network,

$$\boldsymbol{s} = Bilinear\left(\boldsymbol{s}_{global}, \boldsymbol{s}_{local}\right). \tag{15}$$

3.5 Answer Prediction and Evaluation

VideoQA tasks usually come in two forms, multiple-choice and open-ended (Jang et al. 2019; Xu et al. 2017). Given the input video clip $v \in \mathcal{V}$ and the related question $q \in \mathcal{Q}$, multiple-choice task is to choose a correct answer $a^\star$ out of a candidate set $\{a_i\}_{i=1}^{K}$, while open-ended task is to predict an answer $\hat{a}$ belongs to a pre-defined answer set $\mathcal{A}$ that matches the ground answer $a^\star \in \mathcal{A}$.

For multiple-choice task, we follow (Jang et al. 2019) to concentrate the question with each answer candidates so that we obtain K candidate textual sequences. Then we use a linear regression function that inputs the final output vector $\boldsymbol{s}$ and outputs K scores for all candidate answers. We train the whole network by minimizing the hinge loss of pairwise comparisons, scores of incorrect answers $s_i^n, i \leq K-1$ and correct answer s^p, where

$$Loss = \sum_{i=1}^{K-1} \max\left(0, 1 + s_i^n - s^p\right). \tag{16}$$

For open-ended task, we employ a linear classifier and softmax function to predict scores of all answers in $\mathcal{A}$. We train the network by minimizing the cross-entropy loss. Note that for open-ended numbers, the task of repetition counting, we treat it as a regression problem that predict a rounded number (0-10) and adopt ℓ_2 loss to train the network.

4 Experiments

4.1 Experimental Setup

Datasets. We validate the benefits of HGA network on three recent benchmark datasets.

TGIF-QA is a widely used large-scale benchmark dataset for VideoQA (Jang et al. 2017), which consists of 165K Q&A pairs collected from 72K animated GIFs. TGIF-QA defines four task types: (1) Repetition count (*Count*) is an open-ended numbers task to count the number of repetitions of an action in a video; (2) Repeating action (*Action*) is a multiple-choice task to identifying a repetitive action from 5 candidate answers; (3) State transition (*Trans.*) is also a 5-options multiple-choice task which aims to identify the transition of two states; (4) Frame QA (*FrameQA*) is an open-ended task that can be answered from a single frame of a

[1] 4.10 is the mean ℓ_2 loss of rounded numbers, the unrounded value is 4.02 (Fan et al. 2019).

video and this task is formalized as a multi-classification problem aiming at indicating the correct answer from a prepared dictionary.

MSVD-QA and **MSRVTT-QA** are two datasets generated from video descriptions through an automatic method (Xu et al. 2017). There are 50K and 243K Q&A pairs respectively and both consist of five different types of questions, including *what*, *who*, *how*, *when* and *where*. The task is open-ended and aims to identify the answer from a predefined answer set of size 1000.

Implementation details. For fair comparisons with other methods, we follow previous work (Jang et al. 2017; Xu et al. 2017) and take a consistent approach to extract video and text features. Specifically, for TGIF-QA, the pre-trained ResNet-152 and C3D are chosen separately for appearance and motion features, while we use VGG and C3D for MSVD-QA and MSRVTT-QA. For all the three datasets, the pre-trained GloVe word embedding is adopted for text features. These datasets have provided a standard partition of the training, validation and testing sets. In terms of training details, we set the number of the hidden units d to 512. Batch size is set to 64. We use Adam as an optimizer with initial learning rate 10^{-4}. The dropout rate is set to 0.3. For better performance, we use some general training strategies, including early stop, learning rate warming up, and learning rate cosine annealing.

4.2 State of the Art Comparison

TGIF-QA In Table 1, we compare our HGA network against the state-of-the-art methods on TGIF-QA dataset. The following is a brief introduction to these methods:

- The series of ST-VQA (Jang et al. 2017) adopts three different variants of attention-based encoder-decoder model, where "Sp." denotes spatial attention, "Tp." denotes temporal attention and "Sp.Tp." means the joint one.
- CT-SAN trains a concept word detector based on semantic attention for further VideoQA (Yu et al. 2017).
- Co-Mem utilizes a co-memory attention mechanism to integrate motion and appearance features (Gao et al. 2018).
- PSAC uses self-attention instead of RNN to process videos while using a special co-attention (Li et al. 2019).
- Fan et al. proposed heterogeneous memory on appearance and motion features (Fan et al. 2019).
- ST-VQA$\star$ is the current state-of-the-art on TGIF-QA dataset (Jang et al. 2019). It is an extension of the aforementioned ST-VQA series and uses new features and updated model to obtain better performance.

Our HGA network outperforms the most recent ST-VQA$\star$ by 3.1% for *Count*, 2.6% for *Action*, 1.6% for *Trans.* and 6.0% for *FrameQA*. Compared with all methods, HGA network also achieves the best performance in terms of *Count*, *Action* and *Trans.* task and establish new state-of-the-art scores on the three tasks. On *FrameQA* task, we attain comparable result, just 1.1% behind PSAC. The reason for this phenomenon may be due to the model structure and we have

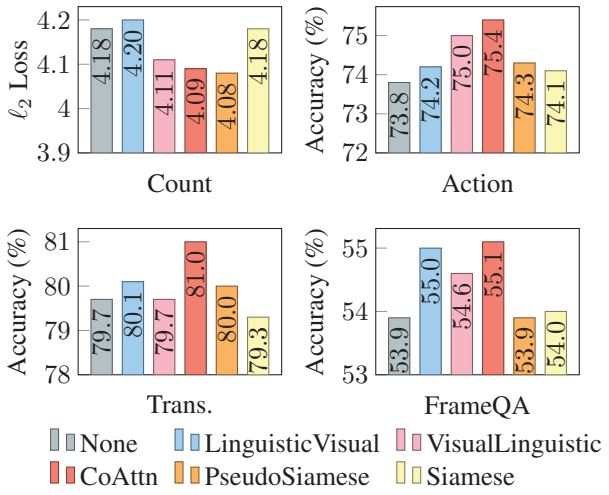

Figure 4: Experimental results of *CAEO* variants.

Table 4: Ablation study on TGIF-QA dataset. Mean ℓ_2 loss for *Count*, and accuracy (%) for others.

Methods	Count	Action	Trans.	FrameQA
GRU (w/ local fusion)	4.31	55.4	69.8	53.0
+ global fusion (baseline)	4.19	73.4	78.1	55.7
+ GCN §3.3	4.18	74.5	79.7	53.9
+ CoAttn §3.2	4.17	73.9	78.8	55.4
+ GCN §3.3	4.09	75.4	81.0	55.1
HGA w/o global fusion	4.25	73.6	79.6	53.6
HGA w/o local fusion	4.24	71.3	77.8	53.9

some observations in the ablation study, discussed in Section 4.3. One aspect to note is that the *Count* result provided by Fan et al. was produced by unrounded ℓ_2 loss. In order to be consistent with other results, we obtain the rounded value through the checkpoint they provided.

MSVD-QA In Table 2, we show a detailed comparison of recent methods to our experimental results on MSVD-QA dataset. ST-VQA, Co-Mem and the work of Fan et al. are described above. AMU (Xu et al. 2017) uses gradually refined attention modules over appearance and motion. The numbers below question types indicate the quantity of such type Q&A pairs in test set and there is a similar distribution in the train set. On the most numerous types, *What* and *Who*, our method outperforms the best previously reported models by 4.9% and 0.6%, and establishes a new state-of-the-art overall accuracy of 34.7% which is almost 3.0% better than the prior best.

MSRVTT-QA In Table 3, we show the experimental results on MSRVTT-QA dataset with most recent contributions. As can be seen from the last column, HGA network achieves a best overall accuracy at 35.5%, outperforming 7.6% than the prior state-of-the-art. The numbers below question types indicate the quantity of such type Q&A pairs in test set. We do better on three question types, *What*, *Who*

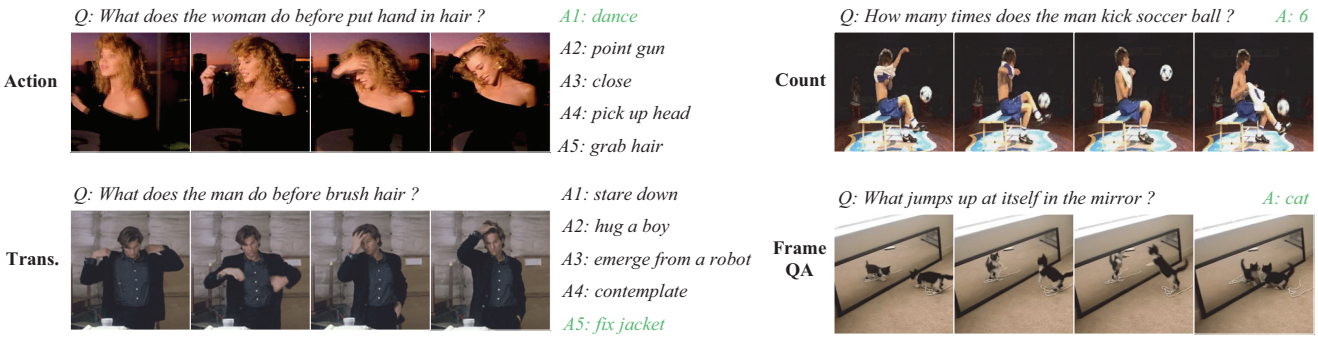

Figure 5: Typical examples of successful cases on TGIF-QA dataset.

and *How*, which account for 99% of all the Q&A pairs.

4.3 Ablation Study

In order to exploit different model variants and analyze the effectiveness of diverse network structures, we conduct in-depth analysis. First, we analyze the effect of different *CAEO* variants. Next, we discuss the ablation study of network structures. We also try to analyze some phenomena encountered in experiments and hope the finding could be useful in informing future research.

TGIF-QA is a proper dataset to establish further analysis on the strengths and limitations of our network. The *Count* task needs to identify actions precisely while the *Action* and *Trans.* task put forward a high demand on temporal reasoning. The *FrameQA* task is not unique to video domain, so we can analyze the relation between VQA and VideoQA. To evaluate the importance of different components, we vary the base network, measuring the change of performance on TGIF-QA dataset.

Analyzing co-attention embedding operations. We first assess how different co-attention embedding operations affect the performance, we fix the other modules and present the results of different variants of *CAEO* in Fig. 4. The first gray bar is shown as a comparison reference where *CAEO* is not used. The remaining five bars correspond to the five schemes we proposed in Section 3.2. We observe that appending the *CAEO* module improve the performance of HGA to some extent. Both linguistic to visual transformation and visual to linguistic transformation have similar single-modality embedding. The former performs better on *Trans.* and *FrameQA*, while the latter is opposite. The pure co-attention transformation, CoAttn, achieves the best performance on *Action*, *Trans.* and *FrameQA*. Although suboptimal on the *Count* task, the difference is very subtle.

We notice that PseudoSiamese and Siamese hurt model quality. This is an interesting phenomenon. In Section 3.2, we explain why *CAEO* is effective through soft KNN theory. Along with this thought, $\boldsymbol{Q}_{CAEO}$, the output of *CAEO*, actually is located in the space where another modality $\boldsymbol{V}$ is. Although $\boldsymbol{Q}_{CAEO}$ and $\boldsymbol{Q}$ have same shapes, there is a huge semantic gap of different modalities between them. The concatenation operator between the two increases inter-modality noise. Using a feed-forward network is still challenging to compensate.

Network variations. We compare some ablation instances of HGA network to investigate the validity of each component, presenting these results in Table 4. In the first line, we give the results of basic single-flow architecture, using GRU to contextualize visual and linguistic sequence with only local fusion. In the second line, we add the global fusion branch, introduced in Section 3.4. So far, we conduct the vanilla parallel architecture, our baseline model. The baseline has shown competitive performance compared to other methods in Table 1, benefiting from the parallel architecture and bilinear fusion modules. Then, we add GCN-based reasoning module to the baseline (+GCN), we report a little gain. Analogously, we add co-attention embedding module to the baseline (+CoAttn), which also boost the performance but subtle. Finally, the GCN is added to the "baseline+CoAttn" and creating the complete HGA network (baseline+CoAttn+GCN). This instance reaches the highest accuracy and there is a significant improvement over others. It is worth noting that the performance of combining "CoAttn" and "GCN" exceeds the sum of the independent two, which indicates the two modules promote each other. In addition, in order to verify the effect of discarding global or local fusion flow, we remove the two branches from the HGA network respectively, revealed in the last two lines which suggests the two fusion flows are both effective.

We also observe that GCN hurts the *FrameQA* performance. *FrameQA* type questions can be answered from one frame in a video. In particular, "What color" is the question far from motion reasoning, while GCN depends on motion clues and shot-level semantic information for reasoning, which is weak on extracting information from single frame.

5 Conclusions

In this paper, we propose HGA network, a novel heterogeneous graph alignment network for VideoQA, which performs four inevitable steps: representation, fusion, alignment, and reasoning. Within HGA, we view multimodal factors, video shots or question words, as nodes in an uniform heterogeneous graph, exploring inter- and intra-modality interactions and cross-modal reasoning. We evaluate our

method on three benchmark datasets and conduct extensive ablation study to the effectiveness of HGA. Experiments show the network to be superior in quality.

References

Ba, J. L.; Kiros, J. R.; and Hinton, G. E. 2016. Layer normalization. *arXiv preprint arXiv:1607.06450*.

Baltrušaitis, T.; Ahuja, C.; and Morency, L.-P. 2018. Multimodal machine learning: A survey and taxonomy. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 41(2):423–443.

Ben-Younes, H.; Cadene, R.; Thome, N.; and Cord, M. 2019. Block: Bilinear superdiagonal fusion for visual question answering and visual relationship detection. *arXiv preprint arXiv:1902.00038*.

Cadene, R.; Ben-Younes, H.; Cord, M.; and Thome, N. 2019. Murel: Multimodal relational reasoning for visual question answering. In *CVPR*, 1989–1998.

Fan, C.; Zhang, X.; Zhang, S.; Wang, W.; Zhang, C.; and Huang, H. 2019. Heterogeneous memory enhanced multimodal attention model for video question answering. In *CVPR*, 1999–2007.

Gao, J.; Ge, R.; Chen, K.; and Nevatia, R. 2018. Motion-appearance co-memory networks for video question answering. In *CVPR*, 6576–6585.

Gao, P.; Jiang, Z.; You, H.; Lu, P.; Hoi, S. C.; Wang, X.; and Li, H. 2019. Dynamic fusion with intra-and inter-modality attention flow for visual question answering. In *CVPR*, 6639–6648.

Goldberger, J.; Hinton, G. E.; Roweis, S. T.; and Salakhutdinov, R. R. 2005. Neighbourhood components analysis. In *Advances in Neural Information Processing Systems*, 513–520.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *CVPR*, 770–778.

Jang, Y.; Song, Y.; Yu, Y.; Kim, Y.; and Kim, G. 2017. Tgif-qa: Toward spatio-temporal reasoning in visual question answering. In *CVPR*, 2758–2766.

Jang, Y.; Song, Y.; Kim, C. D.; Yu, Y.; Kim, Y.; and Kim, G. 2019. Video question answering with spatio-temporal reasoning. *International Journal of Computer Vision* 1385–1412.

Kim, J.; Ma, M.; Kim, K.; Kim, S.; and Yoo, C. D. 2019. Progressive attention memory network for movie story question answering. In *CVPR*, 8337–8346.

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

Lei, J.; Yu, L.; Bansal, M.; and Berg, T. 2018. Tvqa: Localized, compositional video question answering. In *EMNLP*, 1369–1379.

Li, X.; Song, J.; Gao, L.; Liu, X.; Huang, W.; He, X.; and Gan, C. 2019. Beyond rnns: Positional self-attention with co-attention for video question answering. In *AAAI*, 8658–8665.

Nguyen, D.-K., and Okatani, T. 2018. Improved fusion of visual and language representations by dense symmetric co-attention for visual question answering. In *CVPR*, 6087–6096.

Norcliffe-Brown, W.; Vafeias, S.; and Parisot, S. 2018. Learning conditioned graph structures for interpretable visual question answering. In *Advances in Neural Information Processing Systems*, 8334–8343.

Tapaswi, M.; Zhu, Y.; Stiefelhagen, R.; Torralba, A.; Urtasun, R.; and Fidler, S. 2016. Movieqa: Understanding stories in movies through question-answering. In *CVPR*, 4631–4640.

Teney, D.; Liu, L.; and van den Hengel, A. 2017. Graph-structured representations for visual question answering. In *CVPR*, 1–9.

Tran, D.; Bourdev, L.; Fergus, R.; Torresani, L.; and Paluri, M. 2015. Learning spatiotemporal features with 3d convolutional networks. In *CVPR*, 4489–4497.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, 5998–6008.

Wang, X., and Gupta, A. 2018. Videos as space-time region graphs. In *ECCV*, 399–417.

Wang, B.; Xu, Y.; Han, Y.; and Hong, R. 2018. Movie question answering: remembering the textual cues for layered visual contents. In *AAAI*, 7380–7387.

Xu, D.; Zhao, Z.; Xiao, J.; Wu, F.; Zhang, H.; He, X.; and Zhuang, Y. 2017. Video question answering via gradually refined attention over appearance and motion. In *ACM Multimedia*, 1645–1653.

Xue, H.; Chu, W.; Zhao, Z.; and Cai, D. 2018. A better way to attend: attention with trees for video question answering. *IEEE Transactions on Image Processing* 27(11):5563–5574.

Yu, Y.; Ko, H.; Choi, J.; and Kim, G. 2017. End-to-end concept word detection for video captioning, retrieval, and question answering. In *CVPR*, 3165–3173.

Yu, Z.; Yu, J.; Cui, Y.; Tao, D.; and Tian, Q. 2019. Deep modular co-attention networks for visual question answering. In *CVPR*, 6281–6290.

Zhang, D.; Cao, R.; and Wu, S. 2019. Information fusion in visual question answering: A survey. *Information Fusion* 52:268–280.