Aligned Dual Channel Graph Convolutional Network for Visual Question Answering

Qingbao Huang¹,², Jielong Wei², Yi Cai¹*, Changmeng Zheng¹, Junying Chen¹, Ho-fung Leung³, Qing Li⁴
¹School of Software Engineering, South China University of Technology, Guangzhou, China
²School of Electrical Engineering, Guangxi University, Nanning, Guangxi, China
³The Chinese University of Hong Kong, Hong Kong SAR, China
⁴The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong, China
qbhuang@gxu.edu.cn, 1712306010@st.gxu.edu.cn, ycai@scut.edu.cn

Abstract
Visual question answering aims to answer the natural language question about a given image. Existing graph-based methods only focus on the relations between objects in an image and neglect the importance of the syntactic dependency relations between words in a question. To simultaneously capture the relations between objects in an image and the syntactic dependency relations between words in a question, we propose a novel dual channel graph convolutional network (DC-GCN) for better combining visual and textual advantages. The DC-GCN model consists of three parts: an I-GCN module to capture the relations between objects in an image, a Q-GCN module to capture the syntactic dependency relations between words in a question, and an attention alignment module to align image representations and question representations. Experimental results show that our model achieves comparable performance with the state-of-the-art approaches.

1 Introduction
As a form of visual Turing test, visual question answering (VQA) has drawn much attention. The goal of VQA (Antol et al., 2015; Goyal et al., 2017) is to answer a natural language question related to the contents of a given image. Attention mechanisms are served as the backbone of the previous mainstream approaches (Lu et al., 2016; Yang et al., 2016; Yu et al., 2017), however, they tend to catch only the most discriminative information, ignoring other rich complementary clues (Liu et al., 2019).

Recent VQA studies have been exploring higher level semantic representation of images, notably using graph-based structures for better image understanding, such as scene graph generation (Xu et al., 2017; Yang et al., 2018), visual relationship detection (Yao et al., 2018), object counting (Zhang et al., 2018a), and relation reasoning (Cao et al., 2018; Li et al., 2019; Cadene et al., 2019a). Representing images as graphs allows one to explicitly model interactions between two objects in an image, so as to seamlessly transfer information between graph nodes (e.g., objects in an image).

Very recent research methods (Li et al., 2019; Cadene et al., 2019a; Yu et al., 2019) have achieved remarkable performances, but there is still a big gap between them and human. As shown in Figure 1(a), given an image of a group of persons and the corresponding question, a VQA system needs to not only recognize the objects in an image (e.g., batter, umpire and catcher), but also grasp the textual information in the question “what color is the umpire’s shirt”. However, even many competitive VQA models struggle to process them accurately, and as a result predict the incorrect answer (black) rather than the correct answer (blue), including the
Although the relations between two objects in an image have been considered, the attention-based VQA models lack building blocks to explicitly capture the syntactic dependency relations between words in a question. As shown in Figure 1(c), these dependency relations can reflect which object is being asked (e.g., the word *umpire’s* modifies the word *shirt*) and which aspect of the object is being asked (e.g., the word *color* is the direct object of the word *is*). If a VQA model only knows the word *shirt* rather than the relation between words *umpire’s* and *shirt* in a question, it is difficult to distinguish which object is being asked. In fact, we do need the modified relations to discriminate the correct object from multiple similar objects. Therefore, we consider that it is necessary to explore the relations between words at linguistic level in addition to constructing the relations between objects at visual level.

Motivated by this, we propose a dual channel graph convolutional network (DC-GCN) to simultaneously capture the relations between objects in an image and the syntactic dependency relations between words in a question. Our proposed DC-GCN model consists of an Image-GCN (I-GCN) module, a Question GCN (Q-GCN) module, and an attention alignment module. The I-GCN module captures the relations between objects in an image, the Q-GCN module captures the syntactic dependency relations between words in a question, and the attention alignment module is used to align two representations of image and question. The contributions of this work are summarized as follows:

1) We propose a dual channel graph convolutional network (DC-GCN) to simultaneously capture the visual and textual relations, and design the attention alignment module to align the multimodal representations, thus reducing the semantic gaps between vision and language.

2) We explore how to construct the syntactic dependency relations between words at linguistic level via graph convolutional networks as well as the relations between objects at visual level.

3) We conduct extensive experiments and ablation studies on VQA-v2 and VQA-CP-v2 datasets to examine the effectiveness of our DC-GCN model. Experimental results show that the DC-GCN model achieves competitive performance with the state-of-the-art approaches.
Figure 2: Illustration of our proposed Dual Channel Graph Convolutional Network (DC-GCN) for VQA task. The Dependency Parsing constructs the semantic relations between words in a question, and Q-GCN Module updates every word’s features by aggregating the adjacent word features. In addition, the I-GCN Module builds the relations between image objects, and the Attention Alignment Module use question-guided image attention mechanism to learn a new object representation thus align the images and questions. All punctuations and upper cases have been preprocessed. The numbers in red are the weight scores of image objects and words.

3.2 Relation Extraction and Encoding

3.2.1 I-GCN Module

Image Fully-connected Relations Graph By treating each object region in an image as a vertex, we can construct a fully-connected undirected graph, as shown in Figure 3(b). Each edge represents a relation between two object regions. Pruned Image Graph with Spatial Relations Spatial relations represent an object position in an image, which correspond to a 4-dimensional spatial coordinate \([x_1, y_1, x_2, y_2]\). Note that \((x_1, y_1)\) is the coordinate of the top-left point of the bounding box and \((x_2, y_2)\) is the coordinate of the bottom-right point of the bounding box.

Identifying the correlation between objects is a key step. We calculate the correlation between objects by using spatial relations. The steps are as follows: (1) The features of two nodes are input into multi-layer perceptron respectively, and then the corresponding elements are multiplied to get a relatedness score. (2) The intersection over union of two object regions is calculated. According to the overlapping part of two object regions, different spatial relations are classified into 11 different categories, such as inside, cover, and overlap (Yao et al., 2018). Following the work (Yao et al., 2018), we utilize the overlapping region between
two object regions to judge whether there is an edge between two regions. If two object regions have large overlapping part, it means that there is a strong correlation between these two objects. If two object regions haven’t any overlapping part, we consider two objects have a weak correlation, which means there are no edges to connect these two nodes. According to the spatial relations, we prune some irrelevant relations between objects and obtain a sparse graph, as shown in Figure 3(c).

Figure 3: (a) Generate region proposals by pretrained model (Anderson et al., 2018). For display purposes, we only highlight some object regions. (b) Construct the relations between objects. (c) Prune the irrelevant object edges and calculate the weight between objects. The numbers in red are the weights of edges.

### Image Graph Convolutions

Following the previous studies (Li et al., 2019; Zhang et al., 2018b; Yang et al., 2018), we use GCN to update the representations of objects. Given a graph with \( \mu \) nodes, each object region in an image is a node. We represent the graph structure with a \( \mu \times \mu \) adjacency matrix \( A \), where \( A_{ij} = 1 \) if there is overlapping region between node \( i \) and node \( j \); else \( A_{ij} = 0 \).

Given a target node \( i \) and a neighboring node \( j \in \mathcal{N}(i) \) in an image, where \( \mathcal{N}(i) \) is the set of nodes neighboring with node \( i \), and the representations of node \( i \) and node \( j \) are \( h_{vi} \) and \( h_{vj} \), respectively. To obtain the correlation score \( s_{ij} \) between node \( i \) and \( j \), we learn a fully connected layer over concatenated node features \( h_{vi} \) and \( h_{vj} \):

\[
s_{ij} = w_a^T \sigma(W_a[h_{vi}^{(l)}; h_{vj}^{(l)}]),
\]

where \( w_a \) and \( W_a \) are learned parameters, \( \sigma \) is the non-linear activation function, and \( [h_{vi}^{(l)}; h_{vj}^{(l)}] \) denotes the concatenation operation. We apply a softmax function over the correlation score \( s_{ij} \) to obtain weight \( \alpha_{ij} \), as shown in Figure 3(c) where the numbers in red represent the weight scores:

\[
\alpha_{ij} = \frac{\exp(s_{ij})}{\sum_{j \in \mathcal{N}(i)} \exp(s_{ij})}.
\]

The \( l \)-th layer representations of neighboring nodes \( h_{vj}^{(l)} \) are first transformed via a learned linear transformation \( W_b \). Those transformed representations are then gathered with weight \( \alpha_{ij} \), followed by a non-linear function \( \sigma \). This layer-wise propagation can be denoted as:

\[
h_{vi}^{(l+1)} = \sigma \left( h_{vi}^{(l)} + \sum_{j \in \mathcal{N}(i)} A_{ij} \alpha_{ij} W_b h_{vj}^{(l)} \right).
\]

Following the stacked \( L \) layer GCN, the output of I-GCN module \( H_v \) can be denoted as:

\[
H_v = h_{vi}^{(L)} \quad (l < L).
\]

### 3.2.2 Q-GCN Module

In practice, we observe that two words in a sentence usually hold certain relations. Such relations can be identified by the universal Standford Dependencies (De Marneffe et al., 2014). As shown in Table 1, we list a part of commonly-used dependency relations. For example, the sentence “what color is the umpire’s shirt” is parsed to obtain the relations between words (e.g., \( \text{cop} \), \( \text{det} \), \( \text{dobj} \)) are dependency relations. The direction of arrow indicates that two words exist a relation.

The question “the umpire’s shirt” is parsed to obtain the relations between words (e.g., \( \text{cop} \), \( \text{det} \), \( \text{nmod} \)), as shown in Figure 4. The words in blue are the dependency relations. The ending of arrow indicates that this word is a modifier. The word \( \text{root} \) in purple is used to indicate which word is the root node of dependency relations.

### Question Fully-connected Relations Graph

By treating each word in a question as a node, we construct a fully-connected undirected graph, as shown in Figure 5(a). Each edge represents a relation between two words.

### Pruned Question Graph with Dependency Relations

Irrelevant relations between two words may bring noises. Therefore, we need to prune some unrelated relations to reduce the noises. By parsing the dependency relations of a question, we obtain the relations between words (cf. Figure 4). According to dependency relations, we prune some edges between two nodes which do not have dependency relations. A sparse graph is obtained, as shown in Figure 5(b).
Based on the previous works (Gao et al., 2019; Yu et al., 2019), we use self-attention mechanism (Vaswani et al., 2017) to enhance the correlation between words in a question and the correlation between objects in an image, respectively.

To enhance the correlation between words and highlight the important words, we utilize the self-attention mechanism to update question representation $H_q$. The updated question representation $\tilde{H}_q$ is obtained as follows:

$$\tilde{H}_q = \text{softmax} \left( H_q H_q^T \sqrt{d_q} \right) H_q,$$

where $H_q^T$ is the transpose of $H_q$ and $d_q$ is the dimension of $H_q$. The level of this self-attention is set to 4.

To obtain the image representation related to question representation, we align the image representation $H_v$ by utilizing the question representation $\tilde{H}_q$ as the guided vector. The similarity score $r$ between $H_v$ and $\tilde{H}_q$ is calculated as follows:

$$r = \frac{\tilde{H}_q H_v^T}{\sqrt{d_v}},$$

where $H_v^T$ is the transpose of $H_v$ and $d_v$ is the dimension of $H_v$. A softmax function is used to normalize the score $r$ to obtain the weight score $\tilde{r}$:

$$\tilde{r} = [\tilde{r}_1, \cdots, \tilde{r}_i] = \frac{\exp (r_i)}{\sum_{j \in \mu} \exp (r_j)}$$
where $\mu$ is the number of image regions.

By multiplying the weight $\tilde{r}$ and the image representation $\tilde{H}_v$, the updated image representation $\check{H}_v$ is obtained:

$$\check{H}_v = \tilde{r} \cdot H_v.$$  \hspace{1cm} (12)

The level of this question guided image attention is set to 4. The final outputs of the attention alignment module are $\tilde{H}_q$ and $\check{H}_v$.

### 3.4 Answer Prediction

We apply the linear multimodal fusion method to fuse two representations $\tilde{H}_q$ and $\check{H}_v$ as follows:

$$H_r = W^T v \tilde{H}_v + W^T q \tilde{H}_q,$$  \hspace{1cm} (13)

$$\text{pred} = \text{softmax} \left( W_e H_r + b_e \right),$$  \hspace{1cm} (14)

where $W_v, W_q, W_e$, and $b_e$ are learned parameters, and $\text{pred}$ means the probability of the classified answers from the set of answer vocabulary which contains $M$ candidate answers. Following (Yu et al., 2019), we use binary cross-entropy loss function to train an answer classifier.

### 4 Experiments

#### 4.1 Datasets

**VQA-v2** (Goyal et al., 2017) is the most commonly used VQA benchmark dataset which is split into train, val, and test-standard sets. Among test-standard sets, 25% are served as test-dev set. Each question has 10 answers from different annotators. Answers with the highest frequency are treated as the ground truth. All answer types can be divided into Yes/No, Number, and Other. **VQA-CP-v2** (Argawal et al., 2018) is a derivation of the VQA-v2 dataset, which is introduced to evaluate and reduce the question-oriented bias in VQA models. Due to significant difference of distribution between train set and test set, the VQA-CP-v2 dataset is harder than VQA-v2 dataset.

#### 4.2 Experimental Setup

We use the Adam optimizer (Kingma and Ba, 2014) with parameters $\alpha = 0.0001$, $\beta_1 = 0.9$, and $\beta_2 = 0.99$. The size of the answer vocabulary is set to $M=3,129$ as used in (Anderson et al., 2018). The base learning rate is set to 0.0001. After 15 epochs, the learning rate is decayed by $1/5$ every 2 epochs. All the models are trained up to 20 epochs with the same batch size 64 and hidden size 512. Each image has $\mu \in [10, 100]$ object regions, all questions are padded and truncated to the same length 14, i.e., $\lambda = 14$. The levels of stacked layer $L$ and attention alignment module are both 4.

### 4.3 Experimental Results

Table 2 shows the performance of our DC-GCN model and baseline models trained with the widely-used VQA-v2 dataset. All results in our paper are based on single-model performance. For a fair comparison, we also train our model with extra visual genome dataset (Krishna et al., 2017). Bottom-Up compared to train-dev and test-dev set, the VQA-CP-v2 dataset is harder than VQA-v2 dataset.

| Model                  | Test-dev | Test-std |
|------------------------|----------|----------|
|                        | Y/N      | Num      | Other    | All    |
| Bottom-Up              | 81.82    | 44.21    | 56.05    | 65.32  |
| (Anderson et al., 2018)|          |          |          |        |
| DCN (Nguyen and Okatani, 2018) | 83.51 | 46.61    | 57.26    | 66.87  |
| Counter (Zhang et al., 2018a) | 83.14 | 51.62    | 58.97    | 68.09  |
| BAN (Kim et al., 2018)  | 85.31    | 50.93    | 60.26    | 69.52  |
| DFAF (Gao et al., 2019) | 86.09    | 53.32    | 60.49    | 70.22  |
| Erase-Att (Liu et al., 2019) | 85.87 | 50.28    | 61.10    | 70.07  |
| ReGAT (Li et al., 2019) | 86.08    | 54.42    | 60.33    | 70.27  |
| MCAN (Yu et al., 2019)  | 86.82    | 53.26    | 60.72    | 70.63  |

Table 2: Comparison with previous state-of-the-art methods on VQA-v2 test dataset. "-" means data absence. Answer types consist of Yes/No, Num and Other categories. All means the total accuracy rate. All results in our paper are based on single-model performance.

(Anderson et al., 2018) is proposed to use features based on Faster RCNN (Ren et al., 2015) instead of ResNet (He et al., 2016). Dense Co-Attention Network (DCN) (Nguyen and Okatani, 2018) utilizes dense stack of multiple layers of co-attention mechanism. Counting method (Zhang et al., 2018a) is good at counting questions by utilizing the information of bounding boxes. DFAF (Gao et al., 2019) dynamically fuses Intra- and Inter-modality information. ReGAT (Li et al., 2019) models semantic, spatial, and implicit relations via a graph attention network. MCAN (Yu et al., 2019) utilizes deep modular networks to learn the multimodal feature representations, which is a state-of-the-art approach on VQA-v2 dataset. As shown in Table 2, our model increases the overall accuracy of DFAF and MCAN by 1.2% and 0.6% on the test-std set,
Figure 6: Visualizations of the learned attention maps of the Q-GCN module, I-GCN module and Attention Alignment module from some typical layers. We regard the correlation score between nodes as the attention score. Q-GCN(\(l\)) and I-GCN(\(l\)) denote the question GCN attention maps and image GCN attention maps from the \(l\)-th layer, respectively, as shown in (a), (b), (c) and (d). And (e) and (f) mean the question-guided image attention weight of Attention Alignment module in \(l\)-th layer. For the sake of presentation, we only consider 20 object regions in an image. The index within [1, 20] shown on the axes of the attention maps corresponds to each object in the image. For better visualization effect, we highlight in the image three objects which correspond to 4-th, 6-th, 9-th, and 12-th objects, respectively. Although still cannot achieve comparable performance in the category of Num with respect to ReGAT (which is the best one in counting sub-task), our DC-GCN outperforms it in other categories (e.g., Y/N with 1.2%, Other with 1.1% and Overall with 0.9%). It shows that DC-GCN has relation capturing ability in answering all kinds of questions by sufficiently exploring the semantics in both object appearances and object relations. In summary, our DC-GCN achieves outstanding performance on the VQA-v2 dataset.

To demonstrate the generalizability of our DC-GCN model, we also conduct experiments on the VQA-CP-v2 dataset. To overcome the language biases of the VQA-v2 dataset, the research work (Agrawal et al., 2018) designed the VQA-CP-v2 dataset and specifically proposed the GVQA model for reducing the influence of language biases. Table 3 shows the results on VQA-CP-v2 test split. The Murel (Cadene et al., 2019a) and ReGAT (Li et al., 2019) build the relations between objects to realize the reasoning task and question answering task, which are the state-of-the-art models. Our DC-GCN model surpasses both Murel and ReGAT on VQA-CP-v2 (41.47 vs. 39.54 and 41.47 vs. 40.42). The performance gain is lifted to +1.05%. Although our proposed method is not designed for VQA-CP-v2 dataset, our model has a slight adv-

| Model                        | Acc. (%) |
|------------------------------|----------|
| RAMEN (Robik Shrestha, 2019) | 39.21    |
| BAN (Kim et al., 2018) *     | 39.31    |
| Murel (Cadene et al., 2019a)| 39.54    |
| ReGAT-Sem (Li et al., 2019)  | 39.54    |
| ReGAT-Imp (Li et al., 2019)  | 39.58    |
| ReGAT-Spa (Li et al., 2019)  | 40.30    |
| ReGAT (Li et al., 2019)      | 40.42    |
| GVQA (Agrawal et al., 2018) #| 31.30    |
| UpDn (Anderson et al., 2018) **| 39.74 |
| UpDn + Q-Adv + DoE (Ramakrishnan et al., 2018) # | 41.17 |
| DC-GCN (ours)                | 41.47    |

Table 3: Model accuracy on the VQA-CP-v2 benchmark (open-ended setting on the test split). The results of models with * and ** are obtained from the work (Robik Shrestha, 2019) and (Ramakrishnan et al., 2018), respectively. Models with # are designed for solving the language biases. The ReGAT model consists of Semantic (Sem), Implicit (Imp), and Spatial (Spa) relation encoder.

4.4 Qualitative Analysis

In Figure 6, we visualize the learned attentions from the I-GCN module, Q-GCN module and At-
tention Alignment module. Due to the space limitation, we only show one example and visualize six attention maps from different attention units and different layers. From the results, we have the following observations.

**Question GCN Module:** The attention maps of Q-GCN(2) focus on the words *color* and *shirt* as shown in Figure 6(a) while the attention maps of Q-GCN(4) correctly focus on the words *color*, *umpire’s*, and *shirt*, as shown in Figure 6(b). Those words have the larger weight than others. That is to say, the keywords *color*, *umpire’s* and *shirt* are identified correctly.

**Image GCN Module** For the sake of presentation, we only consider 20 object regions in an image. The index within [1, 20] shown on the axes of the attention maps corresponds to each object in the image. Among these indexes, indexes 4, 6, 9, and 12 are the most relevant ones for the question. Compared with I-GCN(2) which focuses on the 4-th, 6-th, 9-th, 12-th, and 14-th objects (cf. Figure 6(c)), the I-GCN(4) focuses more on the 4-th, 6-th, and 12-th objects where the 4-th object has larger weight than the 6-th and 12-th objects, as shown in Figure 6(d). The 4-th object region is the region of ground true while the 6-th, 9-th, and 12-th object regions are the most relevant ones.

**Attention Alignment Module** Given a specific question, a model needs to align image objects guided by the question to update the representations of objects. As shown in Figure 6(e), the focus regions are more scattered, where the key regions are mainly the 4-th, 9-th and 12-th object regions. Through the guidance of the identified words *color*, *umpire’s* and *shirt*, the DC-GCN model gradually pays more attention to the 4-th, 9-th, and 12-th object regions rather than other irrelevant object regions, as shown in Figure 6(f). This alignment process demonstrates that our model can capture the relations of multiple similar objects.

We also visualize some negative examples predicted by our DC-GCN model. As shown in Figure 7, which can be classified into three categories: (1) limitation of object detection; (2) text semantic understanding in scenarios; (3) subjective judgment. In Figure 7(a), although the question *how many sheep are pictured* is not so difficult, the image content is really confusing. If not observe carefully, it’s rather easy to obtain the wrong answer 2 instead of 3. The reasons for this error include object occlusion, near and far degrees, and the limitation of object detection. The image feature extractor is based on Faster R-CNN model (Ren et al., 2015). The accuracy of object detection can indirectly affect the accuracy of feature extraction. Counting subtask in VQA task has a large room to improve. In Figure 7(b), the question *what time should you pay* can be answered by recognizing the text semantic understanding in the image. Text semantic understanding belongs to another task, namely text visual question answering (Biten et al., 2019), which requires to recognize the numbers, symbols and proper nouns in a scene. In Figure 7(c), subjective judgment is needed to answer the question *is this man happy*. Making this judgment requires some common sense knowledge and real life experience. Specifically, someone holding a banana against him and just like holding a gun towards him, so he is unhappy. Our model can not make such analysis like a human being done to make a subjective judgment and predict the correct answer *yes*.

Finally, to understand the distribution of three error types, we randomly pick up 100 samples on dev set of VQA-v2. The number of three error types (i.e., overlapping objects, text semantic understanding, and subjective judgment) is 3, 3, and 29, respectively. The predicted answers of the first two questions types are all incorrect. The last one has 12 incorrect answers, which means the error
rate of this question type is 41.4%. These observations are helpful to make further improvement in the future.

4.5 Ablation Study

We perform extensive ablation studies on the VQA-v2 validation dataset (cf. Table 4). The experimental results are based on one block of our DC-GCN model. All modules inside DC-GCN have the same dimension of 512. The learning rate is 0.0001 and the batch size is 32.

| Component                  | Setting       | Acc. (%) |
|----------------------------|---------------|----------|
| Bottom-Up (Anderson et al., 2018) | Bottom-Up     | 63.15    |
| Default DC-GCN            |               | 66.57    |

| GCN Types                  |               |         |
|---------------------------|---------------|---------|
| DC-GCN                    | 66.57         |         |
| w/o I-GCN                 | 65.52         |         |
| w/o Q-GCN                 | 66.15         |         |

| Dependency relations      |               |         |
|---------------------------|---------------|---------|
| - det                     | 66.50         |         |
| - case                    | 66.42         |         |
| - cop                     | 66.01         |         |
| - aux                     | 66.48         |         |
| - advmod                  | 66.35         |         |
| - compound                | 66.53         |         |
| - det case                | 65.23         |         |
| - det case cop            | 64.11         |         |

Table 4: Ablation studies of our proposed model on VQA-v2 validation dataset. The experimental results are based on one block of our DC-GCN model. w/o means removing a certain module from DC-GCN model. The detailed descriptions about dependency relations are shown on Table 1.

Firstly, we investigate the influence of GCN types. There are two GCN types: I-GCN and Q-GCN, as shown in Table 4. When removing the I-GCN, the performance of our model decreases from 66.57% to 65.52% (p-value = 3.22E-08 < 0.05). When removing the Q-GCN, the performance of our model slightly decreases from 66.57% to 66.15% (p-value = 2.04E-07 < 0.05). We consider that there are two reasons. One is that the image content is more complex than the question’s content, hence which has richer semantic information. By building the relations between objects in an image and the syntactic dependency relations between words in a question. Furthermore, we explicitly construct the relations between words by dependency tree and align the image and question representations by an attention alignment module to reduce the gaps between vision and language. Extensive experiments on the VQA-v2 and VQA-CP-v2 datasets demonstrate that our model achieves comparable performance with the state-of-the-art approaches. We will explore more complicated object relation modeling in future work.

5 Conclusion

In this paper, we propose a dual channel graph convolutional network to explore the relations between objects in an image and the syntactic dependency relations between words in a question. Furthermore, we explicitly construct the relations between words by dependency tree and align the image and question representations by an attention alignment module to reduce the gaps between vision and language. Extensive experiments on the VQA-v2 and VQA-CP-v2 datasets demonstrate that our model achieves comparable performance with the state-of-the-art approaches. We will explore more complicated object relation modeling in future work.

Acknowledgements

We thank the anonymous reviewers for valuable comments and thoughtful suggestions. We would also like to thank Professor Yuzhang Lin from University of Massachusetts Lowell for helpful discussions.

This work was supported by the Fundamental Research Funds for the Central Universities, SCUT (No. 2017ZD048, D2182480), the Science and Technology Planning Project of Guangdong Province (No.2017B050506004), the Science and Technology Programs of Guangzhou (No.201704030076, 201802010027, 201902010046) and the collaborative research grants from the Guangxi Natural Science Foundation (2017GXNSFAA198225) and the Hong Kong Research Grants Council (project no. PolyU 1121417 and project no. C1031-18G), and an internal research grant from the Hong Kong Polytechnic University (project 1.9B0V).
References

Aishwarya Agrawal, Dhruv Batra, Devi Parikh, and Aniruddha Kembhavi. 2018. Don’t just assume; look and answer: Overcoming priors for visual question answering. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 4971–4980.

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, pages 6077–6086.

Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. 2015. Vqa: Visual question answering. In Proceedings of the IEEE international conference on computer vision, pages 2425–2433.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. arXiv preprint arXiv:1409.0473.

Ali Furkan Biten, Ruben Tito, Andres Mafll, Lluis Gomez, Maral Rusiolt, Ernest Valvenny, C. V. Jawahar, and Dimosthenis Karatzas. 2019. Scene text visual question answering. CoRR abs/1905.13648.

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

Remi Cadene, Corentin Dancette, Hedi Ben younes, Matthieu Cord, and Devi Parikh. 2019b. Rubi: Reducing unimodal biases for visual question answering. In Advances in Neural Information Processing Systems, pages 841–852.

Qingxing Cao, Xiaodan Liang, Bailing Li, Guanbin Li, and Liang Lin. 2018. Visual question reasoning on general dependency tree. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 7249–7257.

Marie-Catherine De Marneffe, Timothy Dozat, Natalie a Silveira, Katri Haverinen, Filip Ginter, Joakim Nivre, and Christopher D Manning. 2014. Universal stanford dependencies: A cross-linguistic typology. In LREC, volume 14, pages 4585–4592.

Peng Gao, Zhengkai Jiang, Haoxuan You, Pan Lu, Steven CH Ho, Xiaogang Wang, and Hongsheng Li. 2019. Dynamic fusion with intra-and inter-modality attention flow for visual question answering. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 6639–6648.

Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. 2017. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 6904–6913.

Zhijiang Guo, Yan Zhang, and Wei Lu. 2019. Attention guided graph convolutional networks for relation extraction. In Proceedings of the 57th Conference of the Association for Computational Linguistics (ACL), pages 241–251.

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, pages 770–778.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. Neural computation, 9(8):1735–1780.

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

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.

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

Linjie Li, Zhe Gan, Yu Cheng, and Jingjing Liu. 2019. Relation-aware graph attention network for visual question answering. Proceedings of the IEEE International Conference on Computer Vision, pages 10313–10322.

Xihui Liu, Zihao Wang, Jing Shao, Xiaogang Wang, and Hongsheng Li. 2019. Improving expression grounding with cross-modal attention-guided erasing. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 1950–1959.

Jiasen Lu, Jianwei Yang, Dhruv Batra, and Devi Parikh. 2016. Hierarchical question-image co-attention for visual question answering. In Advances In Neural Information Processing Systems, pages 289–297.

Duy-Kien Nguyen and Takayuki Okatani. 2018. Improved fusion of visual and language representations by dense symmetric co-attention for visual question answering. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 6087–6096.

Marco Pedersoli, Thomas Lucas, Cordelia Schmid, and Jakob Verbeek. 2017. Areas of attention for image captioning. In Proceedings of the IEEE International Conference on Computer Vision, pages 1242–1250.
