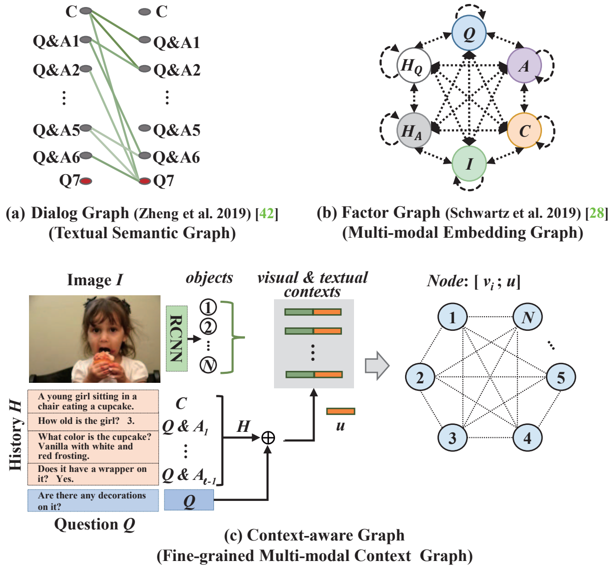
Iterative Context-Aware Graph Inference for Visual Dialog

Dan Guo[1,2] Hui Wang[1,2*] Hanwang Zhang[3] Zheng-Jun Zha[4] Meng Wang[1,2*]
[1]Key Laboratory of Knowledge Engineering with Big Data, Hefei University of Technology
[2]School of Computer Science and Information Engineering, Hefei University of Technology
[3]Nanyang Technological University
[4]University of Science and Technology of China
guodan@hfut.edu.cn, {wanghui.hfut, eric.mengwang}@gmail.com,
hanwangzhang@ntu.edu.sg, zhazj@ustc.edu.cn

Abstract

Visual dialog is a challenging task that requires the comprehension of the semantic dependencies among implicit visual and textual contexts. This task can refer to the relation inference in a graphical model with sparse contexts and unknown graph structure (relation descriptor), and how to model the underlying context-aware relation inference is critical. To this end, we propose a novel Context-Aware Graph (CAG) neural network. Each node in the graph corresponds to a joint semantic feature, including both object-based (visual) and history-related (textual) context representations. The graph structure (relations in dialog) is iteratively updated using an adaptive top-K message passing mechanism. Specifically, in every message passing step, each node selects the most K relevant nodes, and only receives messages from them. Then, after the update, we impose graph attention on all the nodes to get the final graph embedding and infer the answer. In CAG, each node has dynamic relations in the graph (different related K neighbor nodes), and only the most relevant nodes are attributive to the context-aware relational graph inference. Experimental results on VisDial v0.9 and v1.0 datasets show that CAG outperforms comparative methods. Visualization results further validate the interpretability of our method.


C
Q&A1
Q&A2
Q&A5
Q&A6
Q7
C
Q&A1
Q&A2
Q&A5
Q&A6
Q7
H_Q
Q
A
H_A
C
I
Image I
objects
visual & textual contexts
Node: $[v_i; u]$
RCNN
①
②
Ⓝ
History H
A young girl sitting in a chair eating a cupcake.
How old is the girl? 3.
What color is the cupcake? Vanilla with white and red frosting.
Does it have a wrapper on it? Yes.
Are there any decorations on it?
Question Q
C
Q & A_1
Q & $A_{\ell-1}$
H
Q
u
1
N
2
5
3
4

(a) Dialog Graph (Zheng et al. 2019) [42]
(Textual Semantic Graph)

(b) Factor Graph (Schwartz et al. 2019) [28]
(Multi-modal Embedding Graph)

(c) Context-aware Graph
(Fine-grained Multi-modal Context Graph)

Figure 1. Different graph structures for visual dialog. In our solution (c), we focus on a question-conditioned context-aware graph, including both fine-grained visual-objects and textual-history semantics.

1. Introduction

Recently, cross-modal semantic understanding between vision and language has attracted more and more interests, such as image captioning [35, 4, 36, 40, 20], referring expression [11, 41, 21], and visual question answering (VQA) [3, 18, 38, 39]. In these works, the co-reference between vision and language is usually performed in a single round. Taking VQA as an example, given an image and a question, the agent identifies the interest areas related to the question and infers an answer. In contrast, visual dialog [5, 6, 16] is a multi-round extension for VQA. The interactions between the image and multi-round question-answer pairs (history) are progressively changing, and the relationships among the objects in the image are influenced by the current question. Visual dialog is a challenging task due to these underlying semantic dependencies in the textual and visual contexts. Therefore, how to effectively realize the context-aware relational reasoning is vital to the task.

For relational reasoning, the graph structure is employed to exploit the context-aware co-reference of image and history. Except for the graph structure referring to different multi-modal entities as shown in Fig. 1, prior graph-based

[*] Corresponding authors.

models considered the fixed graph attention or embedding, such as *fixed* fully-connected graph (FGA [28]), *fixed* once graph attention evolution (FGA [28]) and *fixed* unidirectional message passing (GNN [42]). In this paper, we are inspired by the nature of the visual dialog task, *i.e.*, the dynamic multi-modal co-references in multi-round conversations. Fig. 2 shows the flexibility and adaptivity of our graph-based method, which iteratively evolves by *adaptive top-K* and *adaptive-directional* message passing. The significance of our method is that it exploits the image-history co-reference in a dynamic adaptive graph learning mode.

In order to comprehend the complex multi-modal co-reference relationships over time, we propose a Context-Aware Graph (CAG) neural network. As shown in Fig. 1 (c), each node in our graph is a multi-modal context representation, which contains both visual objects and textual history contexts; each edge contains the fine-grained visual interactions of the scene objects in the image. In CAG, all the nodes and edges are iteratively updated through an adaptive top-K message passing mechanism. As shown in Fig. 2, in every message passing step, each graph node adaptively selects the most K-relevant nodes, and only receives the messages from them. It means that our CAG solution is an asymmetric dynamic directed graph, which observes adaptive message passing in the graph structure. Note that iterative CAG graph inference is shown as an effective realization of humans' multi-step reasoning [7, 12]. Finally, after the multi-turn graph inference, we impose a graph attention on all the nodes to obtain the final graph embedding for the answer prediction.

Fig. 2 provides an overview of the proposed CAG. Specifically, CAG consists of three components: (1) **Graph Construction** (Sec. 3.1), which constructs the context-aware graph based on the representations of dialog-history and objects in the image; (2) **Iterative Dynamic Directed-Graph Inference** (Sec. 3.2), the context-aware graph is iteratively updated via T-step dynamic directed-graph inference; (3) **Graph Attention Embedding** (Sec. 3.3), which applies a graph attention to aggregate the rich node semantics. Then, we jointly utilize the generated graph, the encoded question, and the history context features to infer the final answer.

The contributions are summarized as follows. We propose a Context-Aware Graph (CAG) neural network for visual dialog, which targets at discovering the partially relevant contexts and building the dynamic graph structure. (1) We build a fine-grained graph representation with various visual objects and attentive history semantics. The context cues on each node not only refer to the joint visual-textual semantic learning, but also involve iterative relational reasoning among image I, question Q and history H. (2) To eliminate the useless relations among the nodes, we design an adaptive top-K message passing mechanism and a graph attention to pick up more relevant context nodes. Each node has different related neighbors (different relations). As for the same node, the inbound and outbound messages vary from iteration to iteration. (3) Extensive experiments are conducted on VisDial v0.9 and v1.0 datasets, and CAG achieves new state-of-the-art performances out of the pervious graph-based methods.

2. Relate Work

Visual Dialog. For the visual dialog task[5, 6, 16], current encoder-decoder based works can be divided into three facets. (1) **Fusion-based models.** Late fusion (LF) [5] and hierarchical recurrent network (HRE) [5] directly encoded the multi-modal inputs and decoded the answer. (2) **Attention-based models**. To improve performance, various attention mechanisms have been widely used in the task, including history-conditioned image attention (HCIAE) [23], sequential co-attention (CoAtt) [34], dual visual attention (DVAN) [9], and recurrent dual attention (ReDAN) [7]. (3) **Visual co-reference resolution models**. Some attention-based works focused on explicit visual co-reference resolution. Seo et al. [29] designed an attention memory (AMEM) to store previous visual attention distrubution. Kottur et al. [15] utilized neural module networks [2] to handle visual co-reference resolution at word-level. Niu et al. [24] proposed a recursive visual attention (RvA) mechanism to recursively reviews history to refine visual attention.

Graph Neural Network (GNN). Graph neural networks have attracted attention in various tasks [32, 22, 17, 8, 26]. The core idea is to combine the graphical structural representation with neural networks, which is suitable for reasoning-style tasks. Liu et al. [31] proposed the first GNN-based approach for **VQA**, which built a scene graph of the image and parsed the sentence structure of the question, and calculated their similarity weights. Later, Norcliffe-Brown et al. [25] modeled a graph representation conditioned on the question, and exploited a novel graph convolution to capture the interactions among different detected object nodes. As for **visual dialog**, there are merely two related works. Zheng et al. [42] proposed an EM-style GNN to conduct the textual co-reference; it regarded the caption and the previous question-answer (QA) pairs as observed nodes, and the current answer was deemed as an unobserved node inferred using EM algorithm on the textual contexts. Schwartz et al. [28] proposed a factor graph attention mechanism, which constructed the graph over all the multi-modal features and estimated their interactions.

Fig. 1 illustrates the difference between our work and other two graph-based models [28, 42]. In this paper, we build a fine-grained context-aware graph, which involves the context co-reference in and between specific objects and history snippets under the guidance of word-level attended question semantics. Apart from the fine-grained object-

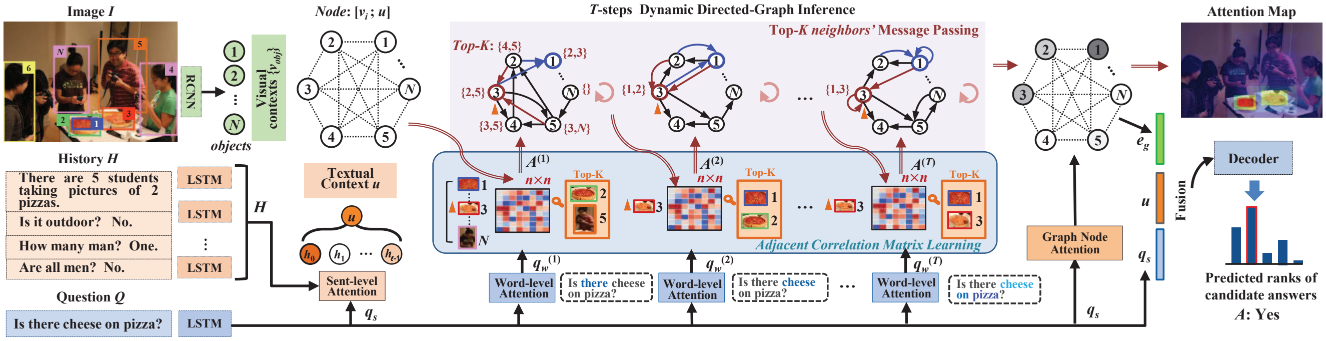


Figure 2. The overall framework of Context-Aware Graph. Our context-aware graph is constructed with visual contexts $\{v_{obj}\}$ and textual context u. The dynamic relations among the nodes are iteratively inferred via Top-K neighbors' Message Passing under the guidance of word-level question command $q_w^{(t)}$. For example, the red and blue nodes in the graph respectively have different top-2 related neighbor nodes, and different directions of the message passing flow on the connected edges.

level features of I (visual contexts) on the node representations, both sentence-level and word-level textual semantics of H and Q are utilized in the graph. We implement iterative dynamic message propagation on edges to aggregate the relationships among nodes for answer prediction. In a nutshell, we realize the cross-modal semantic understanding by context-aware relational reasoning.

3. Our Approach

The visual dialog task refers to relational learning, which involves complicated semantic dependencies among implicit contexts of image, question and history. How to model the context-aware reasoning is critical. In this paper, we propose a dynamic directed-graph inference to iteratively review the multi-modal context cues. Given an image I and the dialog history $H = \{C, (q_1, a_1), ..., (q_{\ell-1}, a_{\ell-1})\}$, where C is the image caption, (q, a) is any question-answer pair and ℓ is the turn number of current dialog. The goal of the model is to infer an exact answer for the current question Q by ranking a list of 100 candidate answers $A = \{a_\ell^{(1)}, ..., a_\ell^{(100)}\}$. The following sub-sections describe the details of the proposed CAG model.

3.1. Graph Construction

Feature Representation. Given an image I, we extract the object-level features using Faster-RCNN [1] and apply a single-layer MLP with activation *tanh* to encode them into a visual feature sequence $V = \{v_1, ..., v_n\} \in \mathbb{R}^{d\times n}$, where n is the number of detected objects. For the current question Q, we first transform it into word embedding vectors $\mathcal{W}^Q = (w_1, ..., w_m) \in \mathbb{R}^{d_w \times m}$, where m denotes the number of tokens in Q. Then we use an LSTM to encode $\mathcal{W}^Q$ into a sequence $U^Q = (h_1^q, ..., h_m^q) \in \mathbb{R}^{d\times m}$, and take the last vector h_m^q as the sentence-level representation of question Q, denoted as $q_s = h_m^q$. Similarly, we adopt another LSTM to extract the features $U^H = (h_0, ..., h_{\ell-1}) \in \mathbb{R}^{d\times \ell}$ of history H at sentence-level, where h_0 is the embedding feature of image caption C.

As questions in a dialog usually have at least one pronoun (*e.g.*, "it", "they", "he"), the dialogue agent is required to discover the relevant textual contexts in the previous history snippets. We employ a question-conditioned attention to aggregate the textual context cues of history, which can be deemed as textual co-reference. The whole process is formulated as follows:

$$\begin{cases} z_h = tanh((W_q q_s)\mathbb{1}^\top + W_h U^H); \\ \alpha_h = softmax(P_h z_h); \\ u = \sum_{j=0}^{\ell-1} \alpha_{h,j} U_j^H, \end{cases} \tag{1}$$

where $W_q, W_h \in \mathbb{R}^{d\times d}$ and $P_h \in \mathbb{R}^{1\times d}$ are learnable parameters, $\mathbb{1} \in \mathbb{R}^{1\times \ell}$ is a vector with all elements set to 1, and $\alpha_{h,j}$ and U_j^H are respective the j-th element of α_h and U^H. $u \in \mathbb{R}^{d\times 1}$ denotes the history-related textual context and is further used to construct the context-aware graph.

Graph Representation. Visual dialog is an on-going conversation. The relations among different objects in the image frequently dynamically vary according to the conversational content. In order to deeply comprehend the conversational content, we build a context-aware graph, which takes both visual and textual contexts into account. The graph structure (relations among objects) will be later iteratively inferred via an adaptive top-K message passing mechanism in Sec. 3.2. Here we construct a graph $G = \{\mathcal{N}, \mathcal{E}\}$, where the i-th node $\mathcal{N}_i$ denotes a joint context feature, corresponding to the i-th visual object feature v_i and its related context feature c_i; the directed edge $\mathcal{E}_{j\to i}$ represents the relational dependency from node $\mathcal{N}_j$ to node $\mathcal{N}_i$ $(i, j \in [1, n])$. Considering the iterative step t, the graph is denoted as

$G^{(t)} = \{\mathcal{N}^{(t)}, \mathcal{E}^{(t)}\}$. There are two cases of $\mathcal{N}^{(t)}$:

$$\begin{cases} \mathcal{N}^{(t)} = (\mathcal{N}_1^{(t)}, ..., \mathcal{N}_n^{(t)}); \\ \mathcal{N}_i^{(t=1)} = [v_i; u]; \ \ \mathcal{N}_i^{(t>1)} = [v_i; c_i^{(t)}], \end{cases} \tag{2}$$

where $[;]$ is the concatenation operation, textual context u is calculated by Eq. 1, and $\mathcal{N}^{(t)} \in \mathbb{R}^{2d \times n}$. For node $\mathcal{N}_i^{(t)}$ in the iterative step t, the visual feature v_i is fixed, and we focus on the context learning of $c_i^{(t)}$.

3.2. Iterative Dynamic Directed-Graph Inference

Visual dialog contains implicit relationships among the image, question and history. From a technical point of view, the key nature of visual dialog is multi-step reasoning and image-history co-reference. Thus, we address it by adaptively capturing the related visual-historical cues in a dynamic multi-modal co-reference mode. Here, we first exploit attended question commands $\{q_w^{(t)}\}$ to instruct message passing on the edges in the graph structure, $t \in [1, T]$, where T is the number of iterations; imitating humans reviewing different keywords multiple times. Then, it is worth noting that in our solution, the "**dynamic directed-graph inference**" process, considers flexible, effective, and relevant message propagation. As shown in Fig. 3, in each iterative step, the context-aware graph is updated through two facets: (1) adjacent correlation learning. Under the instruction of the current question command $q_w^{(t)}$, each node adaptively selects the top-K most relevant nodes as its neighbors based on an adjacent correlation matrix; (2) top-K message passing. To capture the dynamic realtions in the graph, each node receives messages from its top-K neighbors and aggregates these messages to update its context feature.

Question-conditioned Relevance Feedback via Adjacent Correlation Learning. To infer a correct answer, we have to discover accurate semantics of question Q. In each iterative step t, reviewing different words in Q is helpful to locate attentive keywords. Based on the word-level feature sequence of the question $U^Q = (h_1^q, ..., h_m^q)$, we employ a self-attention to obtain the word attention distribution $\alpha_q^{(t)}$. Then, the word embedding sequence $\mathcal{W}^Q = (w_1, ..., w_m)$ is jointly aggregated with $\alpha_q^{(t)}$ to get a new attended question feature $q_w^{(t)}$.

$$\begin{cases} z_q^{(t)} = L2Norm(f_q^{(t)}(U^Q)); \\ \alpha_q^{(t)} = softmax(P_q^{(t)} z_q^{(t)}); \\ q_w^{(t)} = \sum_{j=1}^{m} \alpha_{q,j}^{(t)} w_j, \end{cases} \tag{3}$$

where $f_q^{(t)}(.)$ denotes a two-layer MLP and $P_q^{(t)} \in \mathbb{R}^{1 \times d}$. The parameters of $f_q^{(t)}(.)$ and $P_q^{(t)}$ are independently learned in the t-th iterative step. $q_w^{(t)} \in \mathbb{R}^{d_w \times 1}$ is defined as the t-th question command at the word level.

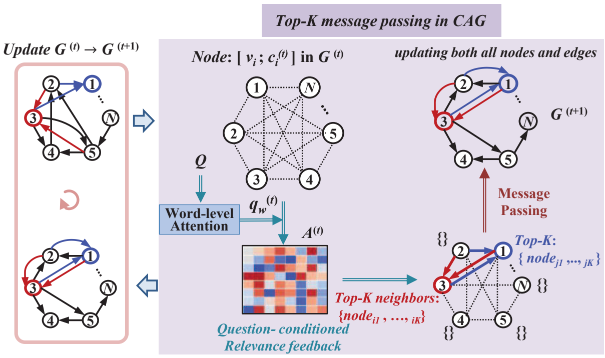

Top-K message passing in CAG
Update $G^{(t)} \rightarrow G^{(t+1)}$
Node: $[v_i; c_i^{(t)}]$ in $G^{(t)}$
updating both all nodes and edges
$G^{(t+1)}$
Q
Word-level Attention
$q_w^{(t)}$
$A^{(t)}$
Message Passing
Top-K: $\{node_{j1}, ..., _{jK}\}$
Top-K neighbors: $\{node_{i1}, ..., _{iK}\}$
Question- conditioned Relevance feedback
1
2
3
4
5
N
{}

Figure 3. Adaptive Top-K Message Passing Unit.

After obtaining the question command $q_w^{(t)}$, we measure the correlation among different nodes in the graph. We design an adjacency correlation matrix of the graph $G^{(t)}$ as $A^{(t)} \in \mathbb{R}^{n \times n}$, in which each value $A_{l \to i}^{(t)}$ represents the connection weight of the edge $\mathcal{E}_{l \to i}^{(t)}$. We learn the correlation matrix $A^{(t)}$ by computing the similarity of each pair of two nodes in $\mathcal{N}^{(t)}$ under the guidance of $q_w^{(t)}$.

$$A^{(t)} = (W_1 \mathcal{N}^{(t)})^\top ((W_2 \mathcal{N}^{(t)}) \odot (W_3 q_w^{(t)})), \tag{4}$$

where $W_1 \in \mathbb{R}^{d \times 2d}$, $W_2 \in \mathbb{R}^{d \times 2d}$ and $W_3 \in \mathbb{R}^{d \times d_w}$ are learnable parameters, and $\odot$ denotes the hadamard product, $i.e.$, element-wise multiplication.

A fact is that there are always only a part of the detected objects in the image related to the question (*i.e.*, sparse relationships). Moreover, among these objects, each object is always irrelated with most of other objects. Therefore, each node in the graph is required to connect with the most relevant neighbor nodes. In order to learn a set of relevant neighbors $S_i^{(t)}$ of node $\mathcal{N}_i^{(t)}$, $i \in [1, n]$, $t \in [1, T]$, we adopt a ranking strategy as:

$$S_i^{(t)} = topK(A_i^{(t)}), \tag{5}$$

where $topK$ returns the indices of the K largest values of an input vector, and $A_i^{(t)}$ denotes the i-th row of the adjacent matrix. Thus, each node has its independent neighbors $S_i^{(t)}$. As shown in Fig. 2 ($topK$, K = 2), even the same node can have different neighbors in different iterative steps. It means that our solution is an adaptive dynamic graph inference process. Our CAG graph is an asymmetric directed-graph.

Relational Graph Learning via Top-K Message Passing. The graph structure is now relational-aware. Each node can be affected by its K neighbor nodes. We propagate the relational cues to each node via message passing. Taking node $\mathcal{N}_i^{(t)}$ in the t-th step as an example, it receives the messages from its most K relevant neighbor nodes $\{\mathcal{N}_j^{(t)}\}$, where $j \in S_i^{(t)}$.

To evaluate the influences of relevant neighbors, $B_{j\to i}^{(t)}$ normalizes the connection weight of edge $\mathcal{E}_{j\to i}^{(t)}$ (*i.e.*, $\mathcal{N}_j^{(t)}$ $\to \mathcal{N}_i^{(t)}$). $A_{j\to i}^{(t)}$ denotes the adjacent correlation weight of edge $\mathcal{E}_{j\to i}^{(t)}$. As shown in Eq. 6, $B_{j\to i}^{(t)}$ normalizes the weights of the neighbour set $\{A_{j\to i}^{(t)}\}$ ($j \in S_i^{(t)}$) with a $softmax$ function. In addition, under the guidance of the question command $q_w^{(t)}$, $m_{j\to i}^{(t)}$ calculates the inbound message of neighbour $\mathcal{N}_j^{(t)}$ to $\mathcal{N}_i^{(t)}$. At last, $\mathcal{N}_i^{(t)}$ sums up all the inbound messages to get the final message feature $M_i^{(t)}$. The whole process is formulated as follows:

$$\begin{cases} [B_{j\to i}^{(t)}]_{j\in S_i^{(t)}} = softmax([A_{j\to i}^{(t)}]); \\ m_{j\to i}^{(t)} = (W_4\mathcal{N}_j^{(t)}) \odot (W_5 q_w^{(t)}); \\ M_i^{(t)} = \sum_{j\in S_i^{(t)}} B_{j\to i}^{(t)} m_{j\to i}^{(t)}, \end{cases} \tag{6}$$

where $W_4 \in \mathbb{R}^{d\times 2d}$ and $W_5 \in \mathbb{R}^{d\times d_w}$ are learnable parameters. $M_i^{(t)} \in \mathbb{R}^{d\times 1}$ denotes the summarized message to $\mathcal{N}_i^{(t)}$, and update $\mathcal{N}_i^{(t)}$ to $\mathcal{N}_i^{(t+1)}$ as follows:

$$\begin{cases} c_i^{(t+1)} = W_6[c_i^{(t)}; M_i^{(t)}]; \\ \mathcal{N}_i^{(t+1)} = [v_i; c_i^{(t+1)}], \end{cases} \tag{7}$$

where $W_6 \in \mathbb{R}^{d\times 2d}$ is a learnable parameter. Parameters $W_1 \sim W_6$ in Eqs. 4 $\sim$ 7 are shared for each iteration. After performing T-step message passing iterations, the final node representation is denoted as $\mathcal{N}^{(T+1)}$.

3.3. Graph Attention Embedding

Up to now, the context learning on each node in the graph $\mathcal{N}^{(T+1)}$ not only integrates original visual and textual features, but also involves iterative context-aware relational learning. As the majority of questions merely pay attention to a small part of objects in the image scene, we apply a question-conditioned graph attention mechanism to attend all the nodes. The graph attention is learned as follows:

$$\begin{cases} z_g = tanh((W_{g_1}q_s)\mathbb{1}^\top + W_{g_2}\mathcal{N}^{(T+1)}); \\ \alpha_g = softmax(P_g z_g); \\ e_g = \sum_{j=1}^{n} \alpha_{g,j}\mathcal{N}_j^{(T+1)}, \end{cases} \tag{8}$$

where $W_{g1} \in \mathbb{R}^{d\times d}$ and $W_{g2} \in \mathbb{R}^{d\times 2d}$ are learnable parameters. $e_g \in \mathbb{R}^{2d\times 1}$ denotes the attended graph embedding.

Finally, we fuse it with the history-related textual context u and the sentence-level question feature q_s to output the multi-modal embedding $\widetilde{e}$:

$$\widetilde{e} = tanh(W_e[e_g; u; q_s]). \tag{9}$$

The output embedding $\widetilde{e}$ is then fed into the discriminative decoder [23] to choose the answer with the highest probability as the final prediction. The details of the training settings are explained in Sec. 4.1.

4. Experiments

4.1. Experiment Setup

Datasets. Experiments are conducted on benchmarks VisDial v0.9 and v1.0 [5]. VisDial v0.9 contains 83k and 40k dialogs on COCO-train and COCO-val images [19] respectively, totally 1.2M QA pairs. VisDial v1.0 is an extension of VisDial v0.9, which adds additional 10k dialogs on Flickr images. The new train, validation, and test splits contains 123k, 2k and 8k dialogs, respectively. Each dialog in VisDial v0.9 consists of 10-round QA pairs for each image. In the test split of VisDial v1.0, each dialog has flexible m rounds of QA pairs, where m is in the range of 1 to 10.

Implementation Details. The proposed method is implemented on the platform of Pytorch. We build the vocabulary that contains the words occurring at least 4 times in the training split. And the captions, questions, and answers are truncated to 40, 20 and 20, respectively. Each word in the dialog is embedded into a 300-dim vector by the GloVe embedding initialization [27]. We adopt Adam optimizer [14] and initialize the learning rate with 4×10^{-4}. The learning rate is multiplied by 0.5 after every 10 epochs. We set all the LSTMs in the model with 1-layer and 512 hidden states, and apply Dropout [30] with ratio 0.3 for attention layers and the last fusion layer. Finally, the model is trained with a multi-class N-pair loss [10, 23].

4.2. Ablation Study of CAG

We evaluate two main hyperparameters in our model CAG – the selected neighbor number K and the number of iterative steps T, and validate the influence of visual features and main components of CAG.

Neighbor Number (K). We test different neighbor numbers $K \in \{1, 2, 4, 8, 16, 36\}$. As shown in Fig. 4, $K = 8$ is an optimal parameter setting. Performances drop significantly for $K < 8$. It means that if the selected neighbor nodes are insufficient, the relational messages can not be fully propagated. While setting the neighbor number $K > 8$, the node receiving redundant irrelevant messages from neighbors can disturb the reasoning ability of the model. Thus, we set the neighbor number $K = 8$ in the following experiments.

Iterative step (T). T indicates the number of relational reasoning steps to arrive the answer. We test different steps T to analysis the influences of iterative inferences. As shown in Table 1, the performance of CAG is gradually improved with the increasing T. We have the best performance while T = 3, lifting R@1 from 53.25 (T = 1) to 54.64. The pro-

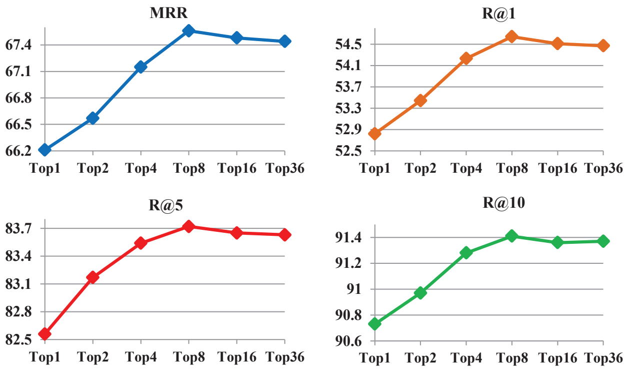


Figure 4. Performance comparison of the neighbor number K on VisDial val v0.9

Model	Step T	Mean↓	MRR↑	R@1↑	R@5↑	R@10↑
CAG	$T = 1$	4.02	66.32	53.25	82.54	90.55
	$T = 2$	3.91	66.93	53.76	83.11	90.96
	$T = 3$	**3.75**	**67.56**	**54.64**	**83.72**	**91.48**
	$T = 4$	3.83	67.28	54.11	83.46	91.17
CAG w/o Infer	-	4.11	65.73	52.56	82.38	90.36
CAG w/o u	$T = 3$	4.19	65.26	51.83	81.55	90.21
CAG w/o Q-att	$T = 3$	3.91	66.70	53.74	82.75	90.89
CAG w/o G-att	$T = 3$	3.86	66.98	53.99	83.08	91.04
CAG	$T = 3$	**3.75**	**67.56**	**54.64**	**83.72**	**91.48**

Table 1. Ablation studies of different iterative steps T and the main components on VisDial val v0.9.

posed iterative graph inference is effective. Visualization results in Fig. 5 further validate this result. When $T > 3$, the performance drops slightly. It means that if the relations in the graph have been fully inferred, further inference does not help. The questions in the VisDial datasets are collected from relative simple free-form human dialogue. The setting of $T = 3$ already performs well. In the following experiment, we set $T = 3$.

Main Component Comparison. A few variants are proposed for ablation study. **CAG w/o Infer** denotes that CAG removes the whole dynamic directed-graph inference in Sec. 3.2. It means that the relations in the graph including all the nodes and edges will not be updated and inferred. **CAG w/o *u*** denotes that CAG without textual-history context u, where the whole graph merely describes visual context cues. **CAG w/o Q-att** denotes CAG without word-level attention on question Q. **CAG w/o G-att** removes the graph attention module, where all the node representations are average pooled to get the final graph embedding.

As shown in Table 1, compared with **CAG**, **CAG w/o Infer** drops MRR significantly from 67.56 to 65.73. It indicates that the graph inference effectively performs well for relational reasoning. Learning the implicit relations among the nodes is helpful to predict the final answer. **CAG w/o *u*** drops R@1 significantly from 54.64 to 51.83. It indicates that the joint visual-textual context learning is necessary. Relations among the nodes can not be fully inferred without textual cues u. **CAG w/o Q-att**, which replaces question commands $\{q_w^{(t)}\}$ with the sentence-level feature q_s, drops R@1 from 54.64 to 53.74. It also can be explained. Figs. 5∼7 demonstrate that the attentive words always vary during the inference process. It usually firstly identifies the target in the question, than focuses on related objects, and finally observes attributes and relations in both visual and textual context cues to infer the answer. **CAG w/o G-att**, which removes the final graph attention module, drops R@1 from 54.64 to 53.99. Although each node involves relational reasoning, not all the nodes are relevant to the current question. Therefore, paying attention to relevant nodes in the relational graph is helpful to infer an exact answer.

Model	Mean↓	MRR↑	R@1↑	R@5↑	R@10↑
		Attention-based Models			
HCIAE [23]	4.81	62.22	48.48	78.75	87.59
AMEM [29]	4.86	62.27	48.53	78.66	87.43
CorefNMN-VGG [15]	4.53	63.60	50.24	79.81	88.51
CoAtt [34]	4.47	63.98	50.29	80.71	88.81
DVAN-VGG [9]	4.38	63.81	50.09	80.58	89.03
RvA-VGG [24]	4.22	64.36	50.40	81.36	89.59
HACAN-VGG [37]	4.32	64.51	50.72	81.18	89.23
		Graph-based Models			
GNN [42]	4.57	62.85	48.95	79.65	88.36
FGA w/o Ans [28]	4.63	62.94	49.35	79.31	88.10
CAG-VGG (Ours)	**4.13**	**64.91**	**51.45**	**81.60**	**90.02**
CAG (Ours)	**3.75**	**67.56**	**54.64**	**83.72**	**91.48**

Table 2. Performance comparison on VisDial val v0.9 with VGG features. Our model with VGG features is denoted as CAG-VGG.

Test with VGG Features. As some existing methods evaluated with VGG features, to be fair, we test our model with VGG features too. Table 2 shows that our **CAG-VGG** still outperforms the previous methods that only utilize VGG features. Compared to **CAG-VGG**, **CAG** gets a significant performance boost. It indicates the object-region features provide richer visual semantics than VGG features.

4.3. Comparison Results

Baseline methods. In our experiment, compared methods can be grouped into three types: (1) **Fusion-based Models** (LF [5] and HRE [5]); (2) **Attention-based Models** (HREA [5], MN [5], HCIAE [23], AMEM [29], CoAtt [34], CorefNMN [15], DVAN [9], RVA [24], Synergistic [10], DAN [13], and HACAN [37]); and (3) **Graph-based Methods** (GNN [42] and FGA [28]).

Results on VisDial v0.9. As shown in Table 3, CAG consistently outperforms most of methods. Compared with fusion-based models **LF** [5] and **HRE** [5], the R@1 performance of our **CAG** is significantly improved, lifting each other by 10.8% and 9.9%. For attention-based models, compared to **DAN** [13], **CAG** outperforms it at all evaluation metrics. **HACAN** [37] reports the recent best results. It first pre-trained with N-pair loss, and then used the wrong answers to "tamper" the truth-history for data augment. Finally, the truth- and fake-history were used to fine-tune its model via reinforcement learning. Without the fine-tuning tactic, our model **CAG** still outperforms **HACAN** on Mean, R@5, and R@10.

Model	VisDial v0.9 (val)					VisDial v1.0 (test-std)					
	Mean↓	MRR↑	R@1↑	R@5↑	R@10↑	Mean↓	NDCG↑	MRR↑	R@1↑	R@5↑	R@10↑
Fusion-based Models											
LF [5]	5.78	58.07	43.82	74.68	84.07	5.95	45.31	55.42	40.95	72.45	82.83
HRE [5]	5.72	58.46	44.67	74.50	84.22	6.41	45.46	54.16	39.93	70.45	81.50
Attention-based Models											
HREA [5]	5.66	58.68	44.82	74.81	84.36	-	-	-	-	-	-
MN [5]	5.46	59.65	45.55	76.22	85.37	5.92	47.50	55.49	40.98	72.30	83.30
HCIAE [23]	4.81	62.22	48.48	78.75	87.59	-	-	-	-	-	-
AMEM [29]	4.86	62.27	48.53	78.66	87.43	-	-	-	-	-	-
CoAtt [34]	4.47	63.98	50.29	80.71	88.81	-	-	-	-	-	-
CorefNMN [15]	4.45	64.10	50.92	80.18	88.81	4.40	54.70	61.50	47.55	78.10	88.80
DVAN [9]	3.93	66.67	53.62	82.85	90.72	4.36	54.70	62.58	48.90	79.35	89.03
RVA [24]	3.93	66.34	52.71	82.97	90.73	4.18	55.59	63.03	49.03	80.40	89.83
Synergistic [10]	-	-	-	-	-	4.17	57.32	62.20	47.90	80.43	89.95
DAN [13]	4.04	66.38	53.33	82.42	90.38	4.30	**57.59**	63.20	49.63	79.75	89.35
HACAN [37]	3.97	**67.92**	**54.76**	83.03	90.68	4.20	57.17	**64.22**	**50.88**	80.63	89.45
Graph-based Models											
GNN [42]	4.57	62.85	48.95	79.65	88.36	4.57	52.82	61.37	47.33	77.98	87.83
FGA w/o Ans [28]	4.63	62.94	49.35	79.31	88.10	-	-	-	-	-	-
FGA [28]	4.35	65.25	51.43	82.08	89.56	4.51	52.10	63.70	49.58	**80.97**	88.55
CAG (Ours)	**3.75**	67.56	54.64	**83.72**	**91.48**	**4.11**	56.64	63.49	49.85	80.63	**90.15**

Table 3. Main comparisons on both VisDial v0.9 and v1.0 datasets using the discriminative decoder [23].

Here, we mainly compare our method with the graph-based models. **GNN** [42] is a recently proposed method, which constructs a graph exploring the dependencies among the textual-history. In contrast, our **CAG** builds a graph over both visual-objects and textual-history contexts. Compared with **GNN**, our model achieves 5.7% improvements on the R@1 metric. **FGA** [28] is the state-of-the-art graph-based method for visual dialog, which treats the candidate answer embedding feature A as new context cue and introduces it into the multi-modal encoding training. This operation improves their results a lot (FGA w/o Ans $vs.$ FGA). Without candidate answer embedding, our model still performs better results, lifting R@1 from 51.43 to 54.64, and decreasing the Mean from 4.35 to 3.75. These comparisons indicate that in our solution, the fine-grained visual-textual joint semantics are helpful for answer inferring.

Results on VisDial v1.0. A new metric NDCG (Normalized Discounted Cumulative Gain) [33] is proposed to evaluate quantitative semantics, which penalizes low ranking correct answers. Other metrics accord to evaluate the rank of the ground-truth in the candidate answer list. NDCG tackles more than one plausible answers in the answer set. Compared with the attention-based models, as above mentioned, **HACAN** [37] trained the model twice and **Synergistic** [10] sorted the candidate answers twice. Without resorting or fine-tuning, under end-to-end training, our model still performs better performance on the Mean value. Compared with the graph-based models, our model has greatly improved the NDCG value. CAG outperforms **GNN** [42] and **FGA** [28] by 3.8% and 4.5% respectively. This also proves that our graph can infer more plausible answers. In addition, we give more intuitive visualization results of CAG to explain the reasoning process detailed in Sec. 4.4.

4.4. Qualitative Results

To further demonstrate the interpretability of our solution, we show an iterative graph inference example in Fig. 5. Two most salient objects ("snowboarder" and "pant") in the graph attention maps are selected to display the inference processes. In iterative step $t = 1$, the question focuses on the word "snowboarder" (target). By reviewing the dialog context, "snowboarder" is related with "midair" and "photographer". The above two salient objects receive messages from their relevant neighbor object nodes. Then in iterative step $t = 2$, the question changes the attention to both words of "snowboarder" and "wearing" (related objects). These two object nodes dynamically update their neighbor nodes under the guidance of the current question command $q_w^{(t=2)}$. In the last step $t = 3$, question Q focuses on the word "wearing" (relation). The edge connections in the graph are further refined by receiving messages from wearing-related nodes. Through multi-step message passing, our context-aware graph progressively finds out much more implicit question-related visual and textual semantics. Finally, the graph attention map overlaying the image I also demonstrates the effectiveness of the graph inference.

We provide another example in Fig. 6 to display relational reasoning in multi-round QA pairs. The edge relations and the nodes' attention weights dynamically vary corresponding to the current question. Our context-aware graph effectively models this dynamic inference process via adaptive top-K message passing. Each node only receives strong messages from the most relevant nodes. The graph attention maps overlaying the image at different rounds further vali-

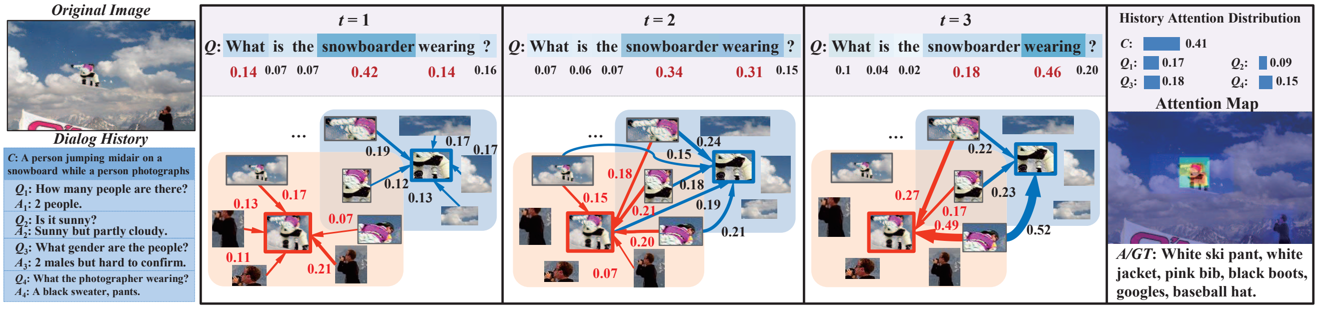

Original Image
Dialog History
C: A person jumping midair on a snowboard while a person photographs
Q_1: How many people are there? A_1: 2 people.
Q_2: Is it sunny? A_2: Sunny but partly cloudy.
Q_3: What gender are the people? A_3: 2 males but hard to confirm.
Q_4: What the photographer wearing? A_4: A black sweater, pants.
t = 1
Q: What is the snowboarder wearing ?
0.14 0.07 0.07 0.42 0.14 0.16
...
0.19 0.17 0.17 0.12 0.13 0.13 0.17 0.07 0.11 0.21
t = 2
Q: What is the snowboarder wearing ?
0.07 0.06 0.07 0.34 0.31 0.15
...
0.24 0.15 0.18 0.18 0.15 0.21 0.19 0.20 0.07 0.21
t = 3
Q: What is the snowboarder wearing ?
0.1 0.04 0.02 0.18 0.46 0.20
...
0.22 0.27 0.17 0.49 0.23 0.52
History Attention Distribution
C: 0.41
Q_1: 0.17
Q_2: 0.09
Q_3: 0.18
Q_4: 0.15
Attention Map
A/GT: White ski pant, white jacket, pink bib, black boots, googles, baseball hat.

Figure 5. Visualization results of iterative context-aware graph inference. It shows the word-level attention on question Q, and dynamic graph inference of the top-2 attended objects (red and blue bounding boxes) in image I. The number on each edge denotes the normalized connection weight, displaying the message influence propagated from neighbors. There are some abbreviations as follows: question (Q), generated answer (A), caption (C) and the ground-truth (GT).

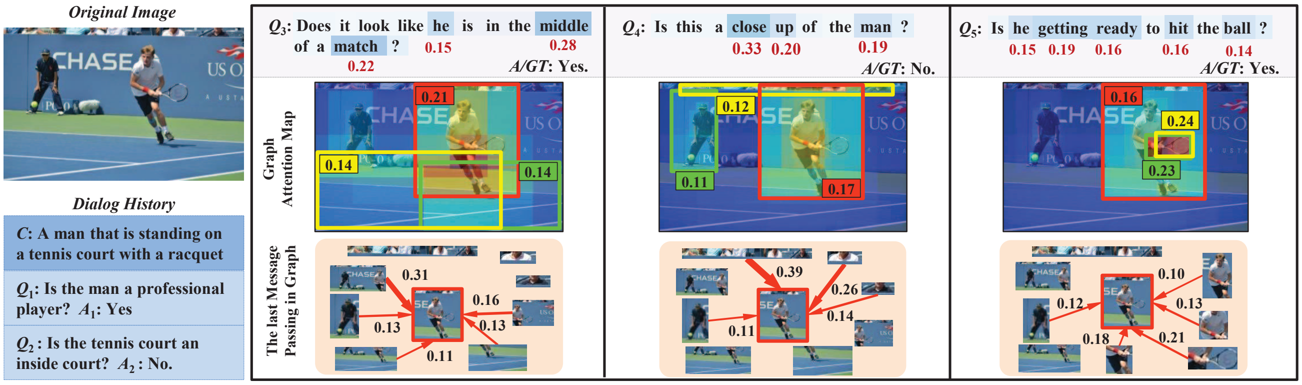

Original Image
Dialog History
C: A man that is standing on a tennis court with a racquet
Q_1: Is the man a professional player? A_1: Yes
Q_2: Is the tennis court an inside court? A_2: No.
Q_3: Does it look like he is in the middle of a match ?
0.15 0.28 0.22
A/GT: Yes.
Q_4: Is this a close up of the man ?
0.33 0.20 0.19
A/GT: No.
Q_5: Is he getting ready to hit the ball ?
0.15 0.19 0.16 0.16 0.14
A/GT: Yes.
Graph Attention Map
0.21 0.14 0.14
0.12 0.11 0.17
0.16 0.24 0.23
The last Message Passing in Graph
0.31 0.16 0.13 0.13 0.11
0.39 0.26 0.14 0.11
0.10 0.12 0.13 0.18 0.21

Figure 6. Visualization result of a progressive multi-round dialog inference. Each column displays the graph attention map overlaying the image I and the last step of message passing process of the most salient object. In these graph attention maps, bounding boxes correspond to the top-3 attended object nodes, and the numbers along with the bounding boxes represent the node attention weights.

date the adaptability of our graph on relational reasoning.

In addition, we display the visualization of attentive word clouds on VisDial v1.0. Fig. 7 describes the word-level attention distribution of question Q in different iterative steps. In iterative step $t = 1$, the proposed CAG inclines to pronouns in the questions, *e.g.*, "there", "it", "you", "they". CAG tries to tackle the textual co-reference in the initial relational reasoning. Then, in step $t = 2$, CAG prefers to attend nouns related to target objects or associated objects in the image, *e.g.*, "people", "building", "tree". This means CAG trends to infer the relationships among different related objects, namely visual-reference. In the time step $t = 3$, the model considers the words that describe the attributes or relations of the objects, *e.g.*, "color", "wearing", "other", "on". All these appearances indicate that we reasonably and actively promote the iterative inference process using the context-aware graph CAG.

5. Conclusion

In this paper, we propose a fine-grained Context-Aware Graph (CAG) neural network for visual dialog, which contains both visual-objects and textual-history context semantics. An adaptive top-K message passing mechanism is proposed to iteratively explore the context-aware representations of nodes and update the edge relationships for a better answer inferring. Our solution is a dynamic directed-graph inference process. Experimental results on the VisDial v0.9 and v1.0 datasets validate the effectiveness of the proposed approach and display explainable visualization results.

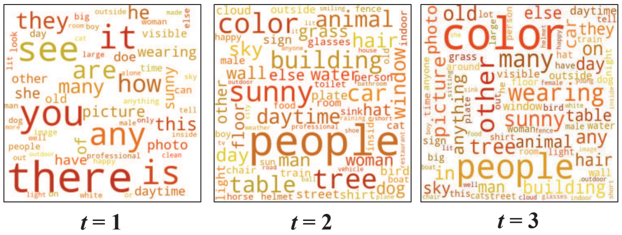


Figure 7. Visualization of attentive word cloud of all the questions $\{Q\}$ in different iterative steps on VisDial v1.0.

Acknowledgements

This work is supported by the National Natural Science Foundation of China (NSFC) under Grants 61725203, 61876058, 61732008, 61622211, and U19B2038.

References

[1] Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. Bottom-up and top-down attention for image captioning and visual question answering. In *CVPR*, pages 6077–6086, 2018. 3

[2] Jacob Andreas, Marcus Rohrbach, Trevor Darrell, and Dan Klein. Neural module networks. In *CVPR*, pages 39–48, 2016. 2

[3] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C. Lawrence Zitnick, and Devi Parikh. Vqa: Visual question answering. In *ICCV*, pages 2425–2433, 2015. 1

[4] Long Chen, Hanwang Zhang, Jun Xiao, Liqiang Nie, Jian Shao, Wei Liu, and Tat-Seng Chua. SCA-CNN: spatial and channel-wise attention in convolutional networks for image captioning. In *CVPR*, pages 6298–6306, 2017. 1

[5] Abhishek Das, Satwik Kottur, Khushi Gupta, Avi Singh, Deshraj Yadav, Jose M. F. Moura, Devi Parikh, and Dhruv Batra. Visual dialog. In *CVPR*, pages 326–335, 2017. 1, 2, 5, 6, 7

[6] Harm de Vries, Florian Strub, Sarath Chandar, Olivier Pietquin, Hugo Larochelle, and Aaron Courville. Guesswhat?! visual object discovery through multi-modal dialogue. In *CVPR*, pages 5503–5512, 2017. 1, 2

[7] Zhe Gan, Yu Cheng, Ahmed El Kholy, Linjie Li, Jingjing Liu, and Jianfeng Gao. Multi-step reasoning via recurrent dual attention for visual dialog. In *ACL*, pages 6463–6474, 2019. 2

[8] Jiuxiang Gu, Handong Zhao, Zhe Lin, Sheng Li, Jianfei Cai, and Mingyang Ling. Scene graph generation with external knowledge and image reconstruction. In *CVPR*, pages 1969–1978, 2019. 2

[9] Dan Guo, Hui Wang, and Meng Wang. Dual visual attention network for visual dialog. In *IJCAI*, pages 4989–4995, 7 2019. 2, 6, 7

[10] Dalu Guo, Chang Xu, and Dacheng Tao. Image-question-answer synergistic network for visual dialog. In *CVPR*, pages 10434–10443, 2019. 5, 6, 7

[11] Ronghang Hu, Huazhe Xu, Marcus Rohrbach, Jiashi Feng, Kate Saenko, and Trevor Darrell. Natural language object retrieval. In *CVPR*, pages 4555–4564, 2016. 1

[12] Drew A. Hudson and Christopher D. Manning. Compositional attention networks for machine reasoning. In *ICLR*, pages 1–1, 2018. 2

[13] Gi-Cheon Kang, Jaeseo Lim, and Byoung-Tak Zhang. Dual attention networks for visual reference resolution in visual dialog. In *EMNLP*, pages 2024–2033, 2019. 6, 7

[14] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2014. 5

[15] Satwik Kottur, Jose M. F. Moura, Devi Parikh, Dhruv Batra, and Marcus Rohrbach. Visual coreference resolution in visual dialog using neural module networks. In *ECCV*, pages 153–169, 2018. 2, 6, 7

[16] Satwik Kottur, José M. F. Moura, Devi Parikh, Dhruv Batra, and Marcus Rohrbach. Clevr-dialog: A diagnostic dataset for multi-round reasoning in visual dialog. In *NAACL*, pages 582–595, 2019. 1, 2

[17] Maosen Li, Siheng Chen, Xu Chen, Ya Zhang, Yanfeng Wang, and Qi Tian. Actional-structural graph convolutional networks for skeleton-based action recognition. In *CVPR*, pages 3595–3603, 2019. 2

[18] Junwei Liang, Lu Jiang, Liangliang Cao, Yannis Kalantidis, Li-Jia Li, and Alexander G. Hauptmann. Focal visual-text attention for memex question answering. *IEEE transactions on pattern analysis and machine intelligence*, 41(8):1893–1908, 2019. 1

[19] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, pages 740–755, 2014. 5

[20] Daqing Liu, Zheng-Jun Zha, Hanwang Zhang, Yongdong Zhang, and Feng Wu. Context-aware visual policy network for sequence-level image captioning. In *ACM MM*, pages 1416–1424, 2018. 1

[21] Xuejing Liu, Liang Li, Shuhui Wang, Zheng-Jun Zha, Dechao Meng, and Qingming Huang. Adaptive reconstruction network for weakly supervised referring expression grounding. In *ICCV*, pages 2611–2620, 2019. 1

[22] Yong Liu, Ruiping Wang, Shiguang Shan, and Xilin Chen. Structure inference net: Object detection using scene-level context and instance-level relationships. In *CVPR*, pages 6985–6994, 2018. 2

[23] Jiasen Lu, Anitha Kannan, Jianwei Yang, Devi Parikh, and Dhruv Batra. Best of both worlds: Transferring knowledge from discriminative learning to a generative visual dialog model. In *NeurIPS*, pages 314–324. 2017. 2, 5, 6, 7

[24] Yulei Niu, Hanwang Zhang, Manli Zhang, Jianhong Zhang, Zhiwu Lu, and Ji-Rong Wen. Recursive visual attention in visual dialog. In *CVPR*, pages 6679–6688, 2019. 2, 6, 7

[25] Will Norcliffe-Brown, Stathis Vafeias, and Sarah Parisot. Learning conditioned graph structures for interpretable visual question answering. In *NeurIPS*, pages 8344–8353, 2018. 2

[26] Y. Peng and J. Chi. Unsupervised cross-media retrieval using domain adaptation with scene graph. *IEEE Transactions on Circuits and Systems for Video Technology*, pages 1–1, 2019. 2

[27] Jeffrey Pennington, Richard Socher, and Christopher Manning. Glove: Global vectors for word representation. In *EMNLP*, pages 1532–1543, 2014. 5

[28] Idan Schwartz, Seunghak Yu, Tamir Hazan, and Alexander G Schwing. Factor graph attention. In *CVPR*, pages 2039–2048, 2019. 2, 6, 7

[29] Paul Hongsuck Seo, Andreas Lehrmann, Bohyung Han, and Leonid Sigal. Visual reference resolution using attention memory for visual dialog. In *NeurIPS*, pages 3719–3729. 2017. 2, 6, 7

[30] Nitish Srivastava, Geoffrey E Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research (JMLR)*, 15(1):1929–1958, 2014. 5

[31] Damien Teney, Lingqiao Liu, and Anton van den Hengel. Graph-structured representations for visual question answering. In *CVPR*, pages 3233–3241, 2017. 2

[32] Peng Wang, Qi Wu, Jiewei Cao, Chunhua Shen, Lianli Gao, and Anton van den Hengel. Neighbourhood watch: Referring expression comprehension via language-guided graph attention networks. In *CVPR*, pages 1960–1968, 2019. 2

[33] Yining Wang, Liwei Wang, Yuanzhi Li, Di He, and Tie-Yan Liu. A theoretical analysis of NDCG type ranking measures. In *COLT*, pages 25–54, 2013. 7

[34] Qi Wu, Peng Wang, Chunhua Shen, Ian Reid, and Anton van den Hengel. Are you talking to me? reasoned visual dialog generation through adversarial learning. In *CVPR*, pages 6106–6115, 2018. 2, 6, 7

[35] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron C. Courville, Ruslan Salakhutdinov, Richard S. Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *ICML*, pages 2048–2057, 2015. 1

[36] Min Yang, Wei Zhao, Wei Xu, Yabing Feng, Zhou Zhao, Xiaojun Chen, and Kai Lei. Multitask learning for cross-domain image captioning. *IEEE Transactions on Multimedia*, 21(4):1047–1061, 2019. 1

[37] Tianhao Yang, Zheng-Jun Zha, and Hanwang Zhang. Making history matter: History-advantage sequence training for visual dialog. In *ICCV*, pages 2561–2569, 2019. 6, 7

[38] Zhou Yu, Jun Yu, Yuhao Cui, Dacheng Tao, and Qi Tian. Deep modular co-attention networks for visual question answering. In *CVPR*, pages 6281–6290, 2019. 1

[39] Zheng-Jun Zha, Jiawei Liu, Tianhao Yang, and Yongdong Zhang. Spatiotemporal-textual co-attention network for video question answering. *The ACM Transactions on Multimedia Computing, Communications, and Applications*, 15(2s):53:1–53:18, 2019. 1

[40] Zhen-Jun Zha, Daqin Liu, Hanwang Zhang, Yongdong Zhang, and Feng Wu. Context-aware visual policy network for fine-grained image captioning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–1, 2019. 1

[41] Fang Zhao, Jianshu Li, Jian Zhao, and Jiashi Feng. Weakly supervised phrase localization with multi-scale anchored transformer network. In *CVPR*, pages 5696–5705, 2018. 1

[42] Zilong Zheng, Wenguan Wang, Siyuan Qi, and Song-Chun Zhu. Reasoning visual dialogs with structural and partial observations. In *CVPR*, pages 6669–6678, 2019. 2, 6, 7