GoG: Relation-aware Graph-over-Graph Network for Visual Dialog

Feilong Chen, Xiuyi Chen, Fandong Meng, Peng Li, Jie Zhou

Pattern Recognition Center, WeChat AI, Tencent Inc, Beijing, China

{ivess.chan,hugheren.chan}@gmail.com
{fandongmeng,patrickpli,withtomzhou}@tencent.com

Abstract

Visual dialog, which aims to hold a meaningful conversation with humans about a given image, is a challenging task that requires models to reason the complex dependencies among visual content, dialog history, and current questions. Graph neural networks are recently applied to model the implicit relations between objects in an image or dialog. However, they neglect the importance of 1) coreference relations among dialog history and dependency relations between words for the question representation; and 2) the representation of the image based on the fully represented question. Therefore, we propose a novel relation-aware graph-over-graph network (GoG) for visual dialog. Specifically, GoG consists of three sequential graphs: 1) H-Graph, which aims to capture coreference relations among dialog history; 2) History-aware Q-Graph, which aims to fully understand the question through capturing dependency relations between words based on coreference resolution on the dialog history; and 3) Question-aware I-Graph, which aims to capture the relations between objects in an image based on fully question representation. As an additional feature representation module, we add GoG to the existing visual dialogue model. Experimental results show that our model outperforms the strong baseline in both generative and discriminative settings by a significant margin.

1 Introduction

Vision-language tasks have drawn more attention with the development of multi-modal natural language processing (Baltrušaitis et al., 2018; Chen et al., 2020b, 2019), such as image captioning (Xu et al., 2015; Anderson et al., 2016, 2018; Cornia et al., 2020; Ghanimifard and Dobnik, 2019), visual question answering (Ren et al., 2015a; Gao et al., 2015; Lu et al., 2016; Anderson et al., 2018;
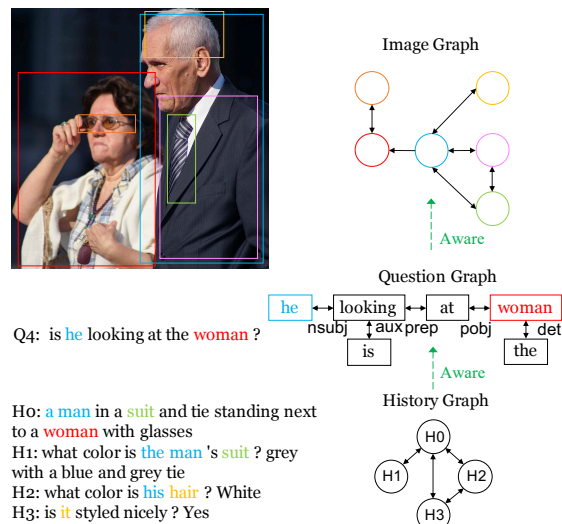

Figure 1: An example of complex relationships in visual dialog. The color in the text corresponds to the same color box in the image, which indicates the same entity. In visual dialog, we construct three graphs. History graph (H-Graph): arrows indicate the coreference relations between QA pairs in dialog history. Question graph (Q-Graph): arrows indicate dependency relations of the question. Image graph (I-Graph): arrows indicate spatial relations between objects in an image. The dark green dotted line indicates the bottom graph affects the upper graph.

Li et al., 2019; Huang et al., 2020) and visual dialog (Das et al., 2017; Kottur et al., 2018; Agarwal et al., 2020; Wang et al., 2020; Qi et al., 2020). Relations in these tasks are significant for reasoning and understanding the textual and visual information. Specifically, visual dialog, which aims to hold a meaningful conversation with a human about a given image, is a challenging task that requires models to reason complex relations among visual content, dialog history, and current questions.

Kinds of attention mechanisms are served as the backbone of previous mainstream approaches (Lu et al., 2017; Wu et al., 2018; Kottur et al., 2018;

Gan et al., 2019; Guo et al., 2019b), following Das et al. 2017. HCAIE (Lu et al., 2017) provides a history-conditioned image attentive encoder to represent the question, the question-attended history, and the attended image. CoAtt (Wu et al., 2018) provides a sequential co-attention encoder to realize that each input feature is co-attended by the other two features in a sequential fashion. ReDAN (Gan et al., 2019) and DMAM (Chen et al., 2020a) use multi-step reasoning based on dual attention to answer a series of questions about an image. DAN (Guo et al., 2019b), MCAN (Agarwal et al., 2020) and LTMI (Nguyen et al., 2020) utilize multi-head attention mechanisms to manage multi-modal intersection. However, these approaches tend to catch only the most discriminative information, ignoring other rich complementary clues, such as relations between objects in an image.

Recent visual dialog studies (Zheng et al., 2019; Schwartz et al., 2019; Jiang et al., 2020b; Guo et al., 2020; Jiang et al., 2020a) explore the higher-level semantic representation of images or dialog history, notably with graph-based structures for modeling the image or dialog history. Although graph-based structures have been considered, these graph-based models lack explicitly capturing complex relations within visual content or textual contexts, and relations between them. As shown in Figure 1, there are complex relations such as coreference relations among dialog history, dependency relations between words in the question, spatial relations between objects in the image. For example, to answer the question Q4 *"is he looking at the woman ?"*, we firstly need to reason in dialog history to know who *"he"* is, then further understand the intention of the question with the understood of history and syntax of questions, and finally know clearly the spatial location and relation about *"the man"* and *"the woman"* in the image based on fully question understanding. How to 1) understand the coreference among history, 2) understand the intention of the question with its syntax and history, 3) understand the image with fully question understanding are worth exploring.

Therefore, in this paper, we propose a novel relation-aware graph-over-graph network (GoG) for visual dialog. Specifically, GoG consists of three sequential graphs: 1) H-Graph, which aims to capture coreference relations among dialog history; 2) History-aware Q-Graph, which aims to fully represent the question through capturing dependency

relations between words based on coreference resolution on the dialog history; and 3) Question-aware I-Graph, which aims to capture the relations between objects in an image on the basis of fully question representation. As an additional feature representation module, we add GoG to the strong visual dialogue model LTMI (Nguyen et al., 2020). We test the effectiveness of our proposed model on two large-scale datasets: VisDial v0.9 and v1.0 (Das et al., 2017). Both automatic and manual evaluations show that our approach can be used to improve the prior strong models. The contributions of this work are summarized as follows:

- We explore how to construct complex explicit relations in visual dialog, i.e., coreference relations among dialog history, dependency relations between words in the question, spatial relations between objects in the image.

- We propose a novel relation-aware graph-over-graph network to reason relations within and among different graphs to obtain a high-level representation of multi-modal information and use it to generate a visually and contextually coherent response.

- We conduct extensive experiments and ablation studies on two large-scale datasets Vis-Dial v0.9 and v1.0. Experimental results show that our GoG model can be used to improve the previous strong visual dialog model in both generative and discriminative settings.

2 Relation-aware Graph-over-Graph Network

2.1 Preliminary

Following Das et al. (Das et al., 2017), a visual dialog agent is given three inputs, i.e., an image I, dialog history (the caption and question-answer pairs) till round $t - 1$: $H = (\underbrace{Cap}_{H_0}, \underbrace{(Q_1, A_1)}_{H_1}, \cdots, \underbrace{(Q_{t-1}, A_{t-1})}_{H_{t-1}})$ and the current question Q_t at round t, where Cap is the caption describing the image taken as H_0, and $H_1, \ldots, H_{t-1}$ are concatenations of question-answer pairs. The goal of the visual dialog agent is to generate a response A_t to the question Q_t.

As shown in Figure 2, our relation-aware graph-over-graph network (GoG) firstly takes the image, the dialog history, and the question as inputs and represent them using Faster RCNN (Ren et al.,
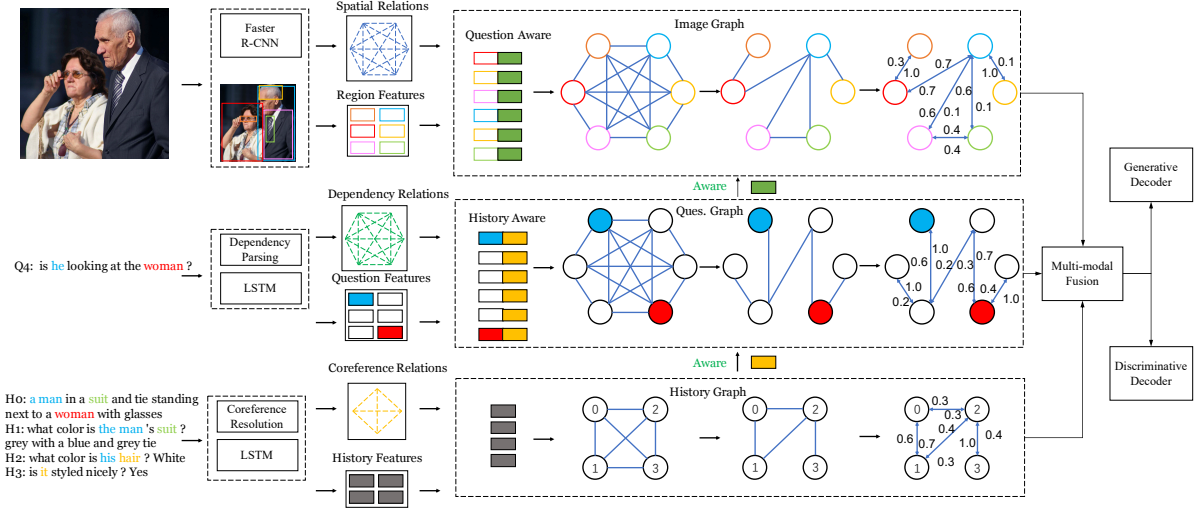
Figure 2: Framework of our Relation-aware Graph-over-Graph Network.

2015b) and LSTM (Hochreiter and Schmidhuber, 1997). Secondly, GoG constructs the history graph, the history-aware question graph, and the question-aware image graph. Thirdly, GoG utilizes the attention alignment module to fuse the three graphs. Finally, GoG uses the fused multi-modal information to give corresponding answers.

Firstly, we simply describe the feature representation of three inputs. Secondly, we introduce our graph attention. Then we describe how we apply our graph attention to the history graph, question graph and image graph to construct our graph-over-graph network. Finally, we describe how we apply our graph-over-graph network to the strong visual dialog models.

2.2 Feature Representation

Similar to (Anderson et al., 2018), we extract the image features by using a pretrained Faster RCNN (Ren et al., 2015b). We select μ object proposals for each image, where each object proposal is represented by a 2048-dimension feature vector. The obtained visual region features are denoted as $v = v_{i=0}^{\mu} \in R^{\mu \times d_v}$.

To extract the question features, each word is embedded into a 300-dimensional vector initialed with the Glove vector (Pennington et al., 2014). The word embeddings are taken as inputs by an LSTM encoder (Hochreiter and Schmidhuber, 1997), which produces the initial question representation $q \in R^{\lambda \times d_q}$. Each history sentence features are obtained as same as the question features. We concatenate the last state $h^{last} \in R^{d_h}$ of each turn history features to get the initial history represen-

tation $h = h_0^{t-1} = [h_0^{last}, \ldots, h_{t-1}^{last}] \in R^{t \times d_h}$. λ denotes the length of the question, t denotes the turn of dialog history, d_q denotes the dimension of each word in questions, d_h denotes the dimension of each word in history and $[\cdot, \cdot]$ denotes the concatenation operation.

2.3 Graph Attention

Given a target node i and a neighboring node $j \in \mathcal{N}(i)$ with a $k \times k$ adjacency matrix R, where $\mathcal{N}(i)$ is the set of k nodes neighboring with node i, and the representations of node i and node j are u_i and u_j, respectively. To obtain the correlation score s_{ij} between node i and j, self-attention (Vaswani et al., 2017) is then performed on the vertices, which generates a relation score s_{ij} between node features u_i and u_j:

$$s_{ij} = \frac{(U_i u_i)^T \cdot V_j u_j}{\sqrt{d_u}}, \qquad (1)$$

where U_i and V_j are trainable parameters. We apply a softmax function over the correlation score s_{ij} to obtain weight α_{ij}:

$$\alpha_{ij} = \frac{\exp(s_{ij} + c_{u,lab(i,j)})}{\sum_{j \in \mathcal{N}(i)} \exp(s_{ij} + c_{u,lab(i,j)})}, \qquad (2)$$

where $c_{\{\cdot\}} = W_{lab} A_{ij}$ is a bias term, $lab(i, j)$ represents the label of each edge, and W_{lab} is a learned parameter. The representations of neighboring nodes u_j are first transformed via a learned linear transformation with W_u. Those transformed representations are then gathered with weight α_{ij}, followed by a non-linear function σ. This propaga-
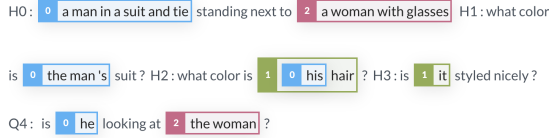
232

Figure 3: The history is performed by coreference resolution. The same color box and the same number indicate the coreference relation between different expressions of the same entity.
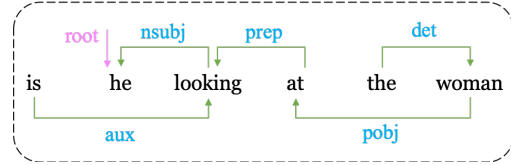

Figure 4: The question is performed by dependency parsing. The word in pink is the root node. The direction of green arrows indicates the dependency relation between two words, and the blue words (e.g., det, dobj) are relation types.

tion can be denoted as:

$$u_i^* = \sigma\Big(u_i + \sum_{j \in \mathcal{N}(i)} R_{ij}\alpha_{ij}W_u u_j\Big). \quad (3)$$

We utilize $\mathrm{GraphAtt}(\cdot)$ to denote equations from Eq. (1) to Eq. (3)

2.4 History Graph Construction

In practice, we observe that coreference relations exist in dialog history. To fully understand the coreference among dialog history, we utilize the coreference resolution tool (Lee et al., 2017) to identify coreference relations. We use the caption and questions to identify the relations instead of QA pairs because there is no ground truth answer in the test split. As shown in Figure 3, we provide a four-turn dialog to show coreference resolution. The same color boxes with the same numbers indicate the coreference relations. For example, the blue box with number 0 indicates they are related to the word "a man" with its attribute "in a suit and tie".

Pruned History Graph with Coreference Relations. We treat each turn history as a node. By analyzing the coreference relations of the history, we obtain the relations between history as shown in Figure 3. According to coreference relations, we construct a sparse graph, as shown in the history graph of Figure 2.

History Graph Attention. Given a graph with t nodes, i.e. a t-turn dialog, each turn representation in history is a node. We represent the graph structure with a $t \times t$ adjacency matrix A, where $A_{ij} = 1$ if there is a coreference relation between node i and node j; else $A_{ij} = 0$.

The relation-aware graph based history representation h_i^* is as follows:

$$h_i^* = \mathrm{GraphAtt}(h_i, A) \quad (4)$$

2.5 Question Graph Construction

In practice, we observe that two words in a sentence usually hold a certain relation. Such relations can be identified by the Neural Dependency Parsing (Dozat and Manning, 2017).

Pruned Question Graph with Dependency Relations. We treat each word in a question as a vertex. By parsing the dependency relations of a question, we obtain the relations between words as shown in Figure 4. According to dependency relations, we obtain our sparse question graph, as shown in the question graph of Figure 2.

History-aware Question Graph Attention. Given a graph with λ nodes, each word in a question is a node. We represent the graph structure with a $\lambda \times \lambda$ adjacency matrix B, where $B_{ij} = 1$ if there is a dependency relation between node i and node j; else $B_{ij} = 0$.

In order to utilize the history to help understand questions, we use a history-aware attention mechanism to inject semantic information from the history into the question graph. The aware history representation is calculated as follows:

$$\hat{h} = \mathrm{softmax}\big(W_{h_1}\sigma(W_{h_2}h^*)\big)h^*, \quad (5)$$

where $\hat{h} \in R^{d_h}$, h^* is the final representation of dialog history. W_{h_1} and W_{h_2} are learned parameters. The history-aware question features are achieved by concatenating the adaptive history representation $\hat{h}$ with each of question features q_i, denoted as:

$$q_i' = [q_i, \hat{h}], \quad \text{for} \quad i = 1, \dots, \lambda. \quad (6)$$

The history-aware and relation-aware graph based question representation q_i^* is as follows:

$$q_i^* = \mathrm{GraphAtt}(q_i', B) \quad (7)$$

2.6 Image Graph Construction

Pruned Image Graph with Spatial Relations. By treating each object region in an image as a vertex, we can construct a fully-connected undirected graph, as shown in the image graph of Figure 2. Each edge represents a relation between two object regions. Spatial relations represent an object position in an image, which correspond to a 4-dimension 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. Following Yao et al. (Yao et al., 2018), we classify different spatial relations into 11 different categories, such as *inside*, *cover* and *overlap*. We utilize the overlapping region between two object regions to judge whether there is an edge between two regions. If two object regions have overlapping parts, it means that there is a strong correlation between these two objects. If two object regions are too far away from each other, it means that there is no relation between these two objects. According to the spatial relations, we prune some irrelevant relations between objects and obtain a sparse graph, as shown in the image graph of Figure 2.

Question-aware Image Graph Attention. Given a graph with μ nodes, each object in an image is a node. We represent the graph structure with a $\mu \times \mu$ adjacency matrix D, where $D_{ij} = 1$ if there is a spatial relation between node i and node j; else $D_{ij} = 0$.

Based on the fully question understanding, we use a question-aware attention mechanism to inject semantic information from the question into the image graph. The adaptive question representation is calculated as follows:

$$\hat{q} = \text{softmax}\big(W_{q_1} \sigma(W_{q_2} q^*)\big) q^*, \qquad (8)$$

where W_{q_1} and W_{q_2} are learned parameters. The question-aware image features are achieved by concatenating the aware question representation $\hat{q}$ with each of the μ image features v_i, denoted as:

$$v_i' = [v_i, \hat{q}], \quad \text{for} \quad i = 1, \dots, \mu. \qquad (9)$$

The question-aware and relation-aware graph based image representation v_i^* is as follows:

$$v_i^* = \text{GraphAtt}(v_i', D) \qquad (10)$$

2.7 Multi-modal Fusion

After obtaining the relation-aware representation, we fuse the question representation q^*, history representation h^*, visual representation v^* through a multi-modal fusion strategy. We can use any existing visual dialog models to learn a joint representation $\mathcal{J}$:

$$\mathcal{J} = \mathcal{F}(q^*, h^*, v^*; \Theta), \qquad (11)$$

where $\mathcal{J}$ is a visual dialog model and Θ are trainable parameters of the fusion module. The design of generative and discriminative decoders (Das et al., 2017), and multi-task learning strategy (Nguyen et al., 2020) can be referred to Appendix A.

3 Experiments

3.1 Experiment Setup

Datasets and Implementation Details. We conduct experiments on the VisDial v0.9 and v1.0 datasets (Das et al., 2017) to verify our approach. VisDial v0.9 contains 83k dialogs on COCO-train (Lu et al., 2017) and 40k dialogs on COCO-val images as the test set, for a total of 1.23M dialog question-answer pairs. VisDial v1.0 dataset is an extension of VisDial v0.9 dataset with additional 10k COCO-like images. VisDial v1.0 dataset contains 123k, 2k, and 8k images as train, validation, and test splits, respectively.

To represent image regions, we use Faster R-CNN (Ren et al., 2015b) with ResNet-101 (He et al., 2016) finetuned on the Visual Genome dataset (Krishna et al., 2017), thus obtaining a 2048-dimension feature vector for each region. Following (Nguyen et al., 2020), we detect K = 100 objects from each image. For the question and history features, we first build the vocabulary composed of 11,322 words that appear at least five times in the training split. The captions, questions, and answers are truncated or padded to 40, 20, and 20 words, respectively. We employ multi-head attention with 4 heads for all three graph attention networks. The dimension of hidden features is set to 512.

Our model is implemented based on Py-Torch (Paszke et al., 2017). In experiments, we use Adam (Kingma and Ba, 2014) optimizer for training, with the mini-batch size as 32. For the choice of the learning rate, we employ the warm-up strategy (Goyal et al., 2017). Specifically, we begin

Model	VisDial v0.9 (val)					VisDial v1.0 (val)					
	MRR ↑	R@1 ↑	R@5 ↑	R@10 ↑	Mean ↓	NDCG ↑	MRR ↑	R@1 ↑	R@5 ↑	R@10 ↑	Mean ↓
Attention-based Model											
RvA (Niu et al., 2019)	55.43	45.37	65.27	72.97	10.71	-	-	-	-	-	-
DVAN (Guo et al., 2019b)	55.94	46.58	65.50	71.25	14.79	-	-	-	-	-	-
DMRM (Chen et al., 2020a)	55.96	46.20	66.02	72.43	13.15	-	50.16	40.15	60.02	67.21	15.19
DAM (Jiang et al., 2020c)	-	-	-	-	-	60.93	50.51	40.53	60.84	67.94	16.65
Pretraining-based Model											
VDBERT (Wang et al., 2020)◇	55.95	46.83	65.43	72.05	13.18	-	-	-	-	-	-
Graph-based Model											
KBGN (Jiang et al., 2020a)	-	-	-	-	-	60.42	50.05	40.40	60.11	66.82	17.54
LTMI (Nguyen et al., 2020)†	55.85	46.07	65.97	72.44	14.17	61.61	50.38	40.30	60.72	68.44	15.73
LTMI-GoG (Ours)	**56.32**	**46.65**	**66.41**	72.69	13.78	**62.63**	**51.32**	**41.25**	**61.83**	**69.44**	**15.32**
LTMI-GoG-Multi (Ours)	56.89	47.04	66.92	72.87	13.45	63.35	51.80	41.78	62.23	69.79	15.16
LTMI-GoG-Multi* (Ours)	59.38	48.58	71.33	78.78	9.94	65.20	55.38	43.93	68.22	76.75	9.98

Table 1: Main comparisons on both VisDial v0.9 and v1.0 datasets using the generative decoder. † denotes that we re-implemented the model. ◇ denotes that the model utilizes large extra datasets for training. Underline indicates the highest performance among previous approaches except pretraining-base models. (t-test, p-value<0.01)

with a learning rate of 0.0001, linearly increasing it at each epoch till it reaches 0.0002 at epoch 4. After 15 epochs, the learning rate is decreased by 1/4 for every 2 epochs up to 20 epochs. We use 4 Titan-XP GPU for training. We spend about 4 hour / 1 epoch for the discriminative setting and 1 hour / 1 epoch for the generative setting. The total parameter of our GoG model is 46.94M, while the total parameter of LTMI (Das et al., 2017) is 42.20M. GoG only has an increase of 4.74M than LTMI.

Automatic Evaluation. We use a retrieval setting to evaluate individual responses at each round of a dialog, following (Das et al., 2017). Specifically, at test time, a list of 100-candidate answers is also given. The model is evaluated on retrieval metrics: (1) Rank of human response, (2) existence of the human response in $top - k$ ranked responses, i.e., R@k (3) Mean reciprocal rank (MRR) of the human response and (4) Normalized discounted cumulative gain (NDCG) for VisDial v1.0.

Human Evaluation. We randomly extract 100 samples for human evaluation (Wu et al., 2018) and then ask 3 human subjects to guess whether the last response in the dialog is human-generated or machine-generated. If at least 2 of them agree it is generated by a human, we think it passes the Truing Test (M1). We record the percentage of responses that are evaluated better than or equal to human responses (M2), according to the human subjects' evaluation.

3.2 Main Results

Baseline methods. In our experiment, compared methods can be grouped into four types: (1) Fusion-based models. (2) Attention-based models: ReDAN, CorefNMN, RvA, DVAN, DMRM, DAM. (3) The pretraining model: VDBERT and VisualBERT. (4) Graph-based models: GNN-EM, DualVD, FGA, KBGN. Please refer to Appendix B for more compared methods.

GoG denotes our relation-aware graph-over-graph network. We use the strong model LTMI (Nguyen et al., 2020)[1] as our multi-modal fusion module. LTMI is a very strong model which achieves some the-state-of-the-art results. "Multi" indicates the model uses multi-task learning at training but utilizes the generative or discriminative decoder at inference, respectively. "Multi*" indicates the model uses multi-task learning and utilizes the discriminative decoder to improve the generative decoder. In general, our model outperforms the strong baseline by a significant margin. We use t-test to analyze our model and LTMI (Nguyen et al., 2020). The p-values is less than 0.01, indicating that the results are significantly different.

Generative Results As shown in the right half of Table 1, we compare generative performance on the val v1.0 split. Our method improves significantly (about 1% on all metrics) on the strong baseline LTMI (Nguyen et al., 2020) and outperforms all the compared methods on all metrics with large margins, which proves that GoG can

---
[1]We reproduce results of LTMI by their official GitHub repo (https://github.com/davidnvq/visdial). We apply the default hyper-parameters as them.

Model	VisDial v0.9 (val)					VisDial v1.0 (test-std)					
	MRR ↑	R@1 ↑	R@5 ↑	R@10 ↑	Mean ↓	NDCG ↑	MRR ↑	R@1 ↑	R@5 ↑	R@10 ↑	Mean ↓
Attention-based Model											
ReDAN (Gan et al., 2019)	-	-	-	-	-	57.63	64.75	51.10	81.73	90.90	3.89
MCA (Agarwal et al., 2020)	-	-	-	-	-	72.73	37.68	20.67	56.67	72.12	8.89
Pretraining-based Model											
VisualBERT (Murahari et al., 2020)◇	-	-	-	-	-	74.47	50.74	37.95	64.13	80.00	6.28
VDBERT (Wang et al., 2020)◇	70.04	57.79	85.34	92.68	4.04	75.35	51.17	38.90	62.82	77.98	6.69
Graph-based Model											
GNN-EM (Zheng et al., 2019)	62.85	48.95	79.65	88.36	4.57	52.82	61.37	47.33	77.98	87.83	4.57
DualVD (Jiang et al., 2020b)	62.94	48.64	80.89	89.94	4.17	56.32	63.23	49.25	80.23	89.70	4.11
FGA (Schwartz et al., 2019)	65.25	51.43	82.08	89.56	4.35	56.90	66.20	52.75	82.92	91.07	3.80
CAG (Guo et al., 2020)	67.56	54.64	83.72	91.48	3.75	56.64	63.49	49.85	80.63	90.15	4.11
KBGN (Jiang et al., 2020a)	-	-	-	-	-	57.60	64.13	50.47	80.70	90.16	4.08
LTMI (Nguyen et al., 2020)†	66.41	53.36	82.53	90.54	4.03	60.74	61.20	47.08	77.78	87.60	4.88
LTMI-GoG (Ours)	66.76	53.84	82.89	90.90	3.91	60.38	63.13	49.88	79.65	89.05	4.39
LTMI-GoG-Multi (Ours)	66.97	54.03	83.10	91.22	3.83	61.04	63.52	50.01	80.13	89.28	4.31

Table 2: Main comparisons on both VisDial v0.9 and v1.0 datasets using the discriminative decoder. ◇ denotes that the model utilizes large extra datasets for training. Underline indicates the highest performance among previous approaches except the pretraining-based models. (t-test, p-value<0.01)

Model	NDCG	MRR	R@1	R@5	R@10	Mean
ReDAN (Gan et al., 2019)	-	64.29	50.65	81.29	90.17	4.10
KBGN	59.08	64.86	51.37	81.71	90.54	4.00
VDBERT (Wang et al., 2020)‡	56.20	62.25	48.16	79.57	89.01	4.31
VDBERT (Wang et al., 2020)◇	63.22	67.44	54.02	83.96	92.33	3.53
LTMI (Nguyen et al., 2020)†	61.52	62.31	48.92	78.55	87.77	4.86
LTMI-GoG	**62.24**	63.81	50.33	80.48	89.24	4.35
LTMI-GoG-Multi	63.15	62.68	49.46	78.77	87.87	4.81

Table 3: Main comparisons on VisDial v1.0 val datasets using the discriminative decoder.

Row	Model	NDCG
0	LTMI	61.61
	LTMI-GoG	62.63
1	w/o I-Graph	61.96
2	w/o Q-Graph	62.15
3	w/o H-Graph	62.03
4	w/o Q-Aware	62.41
5	w/o H-Aware	62.31
6	w/o Spatial Relation	62.15
7	w/o Dependency Relation	62.24
8	w/o Coreference Relation	62.31

Table 4: Ablation study on VisDial v1.0 val datasets using the generative decoder.

improve the performance of visual dialog models by introducing explicit relation reasoning. Compared with the graph-based model KBGN (Jiang et al., 2020a), our GoG-gen improves NDCG from 60.42 to 62.63 (+2.21%), MMR from 50.05 to 51.32 (+1.27%), which illustrates that our explicit relation reasoning is more effective because our approach reduce the noise of implicit relation modeling. LTMI-GoG-Multi and LTMI-GoG-Multi* obtain higher performance with large margins comparing with LTMI (Nguyen et al., 2020), which shows that our approach is effective on multi-task setting. As shown in the left half of Table 1, we come to a similar conclusion on the val v0.9 split. Our method improves a big margin (about 0.5% on all metrics) on LTMI (Nguyen et al., 2020) and outperforms all the none pre-trained methods on MRR, R@1, and R@5.

Discriminative Results As shown in the right half of Table 2, our method improves a lot (near 1.5% on all metrics except NDCG) based on LTMI (Nguyen et al., 2020) on the test-std v1.0 split. We also compare the performance on the val v1.0 split as shown in Table 3. As shown in the left

half of Table 2, we compare discriminative performance on the val v0.9 split. Our method improves a lot based on the LTMI (Nguyen et al., 2020). As shown in Table 3, our approach outperforms VDBERT (Wang et al., 2020)‡ which trains from scratch without extra datasets. All the comparison show that our approach is valid due to explicit relation modeling.

3.3 Ablation Study

As shown in Table 4, we firstly remove the I-Graph, Q-Graph, H-Graph to validate the effect of each graph, respectively. Secondly, we validate the importance of concatenating operation. Finally, we use full connections to replace the relation in the graph to validate the importance of each relation. Firstly, the comparison between line 0 and line 1/2/3 shows all three graphs are crucial for visual dialog, leading to higher performance, and the I-Graph is most important. Secondly, the comparison between line 0 and line 4/5 shows that adding adaptive features gives a gain of approximately

	LTMI (Nguyen et al., 2020)	GoG
Method 1 (M1)	53	64
Method 2 (M2)	60	67

Table 5: Human evaluation on 100 sampled responses on VisDial val v1.0. M1: percentage of responses pass the Turing Test. M2: percentage of responses evaluated better than or equal to human responses.

about +0.2. Thirdly, the comparison between line 1 and line 6/7/8 shows that doing graphs with relations gives better gain than simple fully-connected graphs. Spatial relation is the pick of the bunch because the full connection of 100 objects in an image will bring lots of noise.

3.4 Human Study

As shown in Table 5, we conduct human study to further prove the effectiveness of our model. Our model achieves the highest scores both on the metric M1 and M2 compared with the previous model, LTMI (Nguyen et al., 2020). These results show that our model can generate a better contextually and visually coherent response.

3.5 Qualitative Results

As shown in Figure 5, we visualize the learned attention maps. For the image, the colorful region means higher attention weights. We draw the bounding boxes of the first three highest scores. For the question, the word which has the darker color has higher attention weights. For dialog history, the darker QA pairs have a higher coreference score with the question. Figure 6 provides some dialog examples, as a comparison of the results between GoG and the state-of-the-art LTMI model in the supplementary material. We have two observations by analyzing a set of randomly selected examples. Firstly, GoG generally provides more accurate answers. Secondly, GoG tends to provide longer and more natural human-like answers. More examples can be referred to Appendix B.

4 Related Work

4.1 Visual dialog

For the visual dialog task (Das et al., 2017), GNN-EM (Zheng et al., 2019) utilizes an EM-style GNN to conduct the textual coreference, which regards the caption and the previous question-answer (QA) pairs as observed nodes, and the current answer is deemed as an unobserved node inferred using

EM algorithms (Moon, 1996) on the textual contexts. FGA (Schwartz et al., 2019) realizes a factor graph attention mechanism, which constructs the graph over all the multi-modal features and estimates their interactions. DualVD (Jiang et al., 2020b) constructs a scene graph to represent the image while embedding both relationships provided by (Zhang et al., 2019b) and original object detection features (Anderson et al., 2018). CAG (Guo et al., 2020) focuses on an iterative question-conditioned context-aware graph, including both fine-grained visual-objects and textual-history semantics. In this paper, we model explicit complex relations within and among visual content, dialog history and the current question and design a graph-over-graph structure which are different from graph-based models mentioned above.

4.2 Graph Neural Network

Graph neural networks (Kipf and Welling, 2016; Veličković et al., 2017; Xinyi and Chen, 2018; Zhang et al., 2019a) have attracted attention in various tasks (Wang et al., 2019; Liu et al., 2018; Gu et al., 2019). The core idea is to combine the graphical structural representation with neural networks, which is suitable for reasoning-style tasks. For visual question answering, Liu et al. (Teney et al., 2017) propose the first GNN-based approach, which builds a scene graph of the image and parses the sentence structure of the question, and calculates their similarity weights. Li et al. (Li et al., 2019) propose to encode each image into a graph and model multi-type inter-object relations via a graph attention mechanism, such as spatial relations and semantic, and implicit relations (Li et al., 2019). Huang et al. (Huang et al., 2020) propose a novel dual-channel graph convolutional network for better combining visual and textual advantages. These approaches are limited to built independent graphs. There is no exploration of the coreference among dialog history and relations between graphs in the approach mentioned above.

5 Conclusion

In this paper, we present a relation-aware graph-over-graph network (GoG), a novel framework for visual dialog, which models and reasons the explicit complex relations among visual content, dialog history, and the current question. GoG exploits the graph-over-graph structure to obtain three relation-aware multi-modal representation which
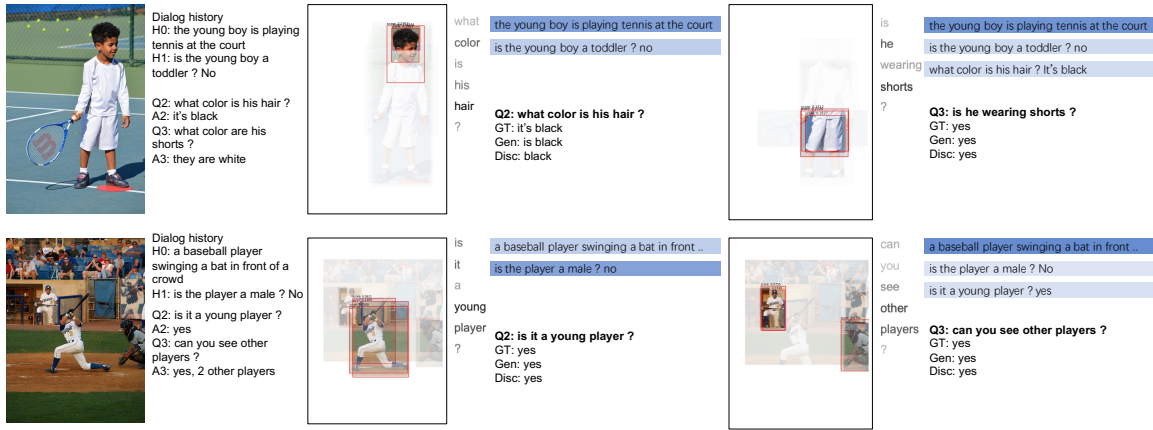
Figure 5: Visualization of attention maps generated in our model at two Q&A rounds on two images.

traffic light in front a tree a statue of a lion is behind the tree a large building is in back

Question	Ground Truth	Generation by GoG	Generation by LTMI	Retrieval by GoG	Retrieval by LTMI
Q1: are there a lot of tree ?	i see 1 tree and some bushes	i see one tree	just see 1 tree	i see 1 tree and some bushes	no, just 1
Q2: what color is the traffic light ?	it 's red	it 's red	red 's a	the light is red	the light is red
Q3: is this in black and white ?	it 's in color	no, it 's in color	no 's a color	it 's in color	no
Q4: is there more than 1 building in the background ?	i can only see one	i only only see 1	no see only see the building	i can only see one	i can only see one
Q5: are there animals ?	no live animals	no	no	no live animals	no live animals

Figure 6: Examples of dialogs generated and retrieved by our model and the LTMI baseline. Our model provides answers that are more accurate than LTMI (green denotes correct answers, and red denotes wrong answers). Results from our model are also more natural and comprehensive (highlighted in blue).

can be added to prior visual dialog models. Experimental results on two large-scale datasets show that our approach improves the previous models by a significant margin.

References

Shubham Agarwal, Trung Bui, Joon-Young Lee, Ioannis Konstas, and Verena Rieser. 2020. History for visual dialog: Do we really need it? *arXiv preprint arXiv:2005.07493*.

Peter Anderson, Basura Fernando, Mark Johnson, and Stephen Gould. 2016. SPICE: Semantic propositional image caption evaluation. *Adaptive Behavior*, 11(4):382–398.

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.

Tadas Baltrušaitis, Chaitanya Ahuja, and Louis-Philippe Morency. 2018. Multimodal machine learning: A survey and taxonomy. *IEEE transactions on pattern analysis and machine intelligence*, 41(2):423–443.

Feilong Chen, Fandong Meng, Jiaming Xu, Peng Li, Bo Xu, and Jie Zhou. 2020a. DMRM: A dual-channel multi-hop reasoning model for visual dialog. *Thirty-Fourth AAAI Conference on Artificial Intelligence*.

Xiuyi Chen, Fandong Meng, Peng Li, Feilong Chen, Shuang Xu, Bo Xu, and Jie Zhou. 2020b. Bridging the gap between prior and posterior knowledge selection for knowledge-grounded dialogue generation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3426–3437.

Xiuyi Chen, Jiaming Xu, and Bo Xu. 2019. A working memory model for task-oriented dialog response generation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2687–2693.

Marcella Cornia, Matteo Stefanini, Lorenzo Baraldi, and Rita Cucchiara. 2020. Meshed-memory transformer for image captioning. In *Proceedings of the*

IEEE/CVF Conference on Computer Vision and Pattern Recognition.

Abhishek Das, Satwik Kottur, Khushi Gupta, Avi Singh, Deshraj Yadav, José MF Moura, Devi Parikh, and Dhruv Batra. 2017. Visual dialog. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 326–335.

Timothy Dozat and Christopher D Manning. 2017. Deep biaffine attention for neural dependency parsing. In *International Conference on Learning Representations*.

Zhe Gan, Yu Cheng, Ahmed EI Kholy, Linjie Li, Jingjing Liu, and Jianfeng Gao. 2019. Multi-step reasoning via recurrent dual attention for visual dialog. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6463–6474.

Haoyuan Gao, Junhua Mao, Jie Zhou, Zhiheng Huang, Lei Wang, and Wei Xu. 2015. Are you talking to a machine? dataset and methods for multilingual image question. In *Advances in Neural Information Processing Systems*, pages 2296–2304.

Mehdi Ghanimifard and Simon Dobnik. 2019. What goes into a word: generating image descriptions with top-down spatial knowledge. In *Proceedings of the 12th International Conference on Natural Language Generation*, pages 540–551.

Priya Goyal, Piotr Dollár, Ross Girshick, Pieter Noordhuis, Lukasz Wesolowski, Aapo Kyrola, Andrew Tulloch, Yangqing Jia, and Kaiming He. 2017. Accurate, large minibatch sgd: Training imagenet in 1 hour. *arXiv preprint arXiv:1706.02677*.

Jiuxiang Gu, Handong Zhao, Zhe Lin, Sheng Li, Jianfei Cai, and Mingyang Ling. 2019. Scene graph generation with external knowledge and image reconstruction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1969–1978.

Dalu Guo, Chang Xu, and Dacheng Tao. 2019a. Image-question-answer synergistic network for visual dialog. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 10434–10443.

Dan Guo, Hui Wang, and Meng Wang. 2019b. Dual visual attention network for visual dialog. pages 4989–4995.

Dan Guo, Hui Wang, Hanwang Zhang, Zheng-Jun Zha, and Meng Wang. 2020. Iterative context-aware graph inference for visual dialog. *arXiv preprint arXiv:2004.02194*.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. *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.

Qingbao Huang, Jielong Wei, Yi Cai, Changmeng Zheng, Junying Chen, Ho-fung Leung, and Qing Li. 2020. Aligned dual channel graph convolutional network for visual question answering. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7166–7176.

Xiaoze Jiang, Siyi Du, Zengchang Qin, Yajing Sun, and Jing Yu. 2020a. KBGN: Knowledge-bridge graph network for adaptive vision-text reasoning in visual dialogue. *Proceedings of the 28th ACM International Conference on Multimedia*.

Xiaoze Jiang, Jing Yu, Zengchang Qin, Yingying Zhuang, Xingxing Zhang, Yue Hu, and Qi Wu. 2020b. DualVD: An adaptive dual encoding model for deep visual understanding in visual dialogue. In *AAAI*, volume 1, page 5.

Xiaoze Jiang, Jing Yu, Yajing Sun, Zengchang Qin, Zihao Zhu, Yue Hu, and Qi Wu. 2020c. DAM: Deliberation, abandon and memory networks for generating detailed and non-repetitive responses in visual dialogue. *arXiv preprint arXiv:2007.03310*.

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

Thomas N Kipf and Max Welling. 2016. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*.

Satwik Kottur, José M. F. Moura, Devi Parikh, Dhruv Batra, and Marcus Rohrbach. 2018. Visual coreference resolution in visual dialog using neural module networks. *ArXiv*, abs/1809.01816.

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

Kenton Lee, Luheng He, Mike Lewis, and Luke Zettlemoyer. 2017. End-to-end neural coreference resolution. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 188–197.

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

Yong Liu, Ruiping Wang, Shiguang Shan, and Xilin Chen. 2018. Structure inference net: Object detection using scene-level context and instance-level relationships. In *Proceedings of the IEEE conference*

on computer vision and pattern recognition, pages 6985–6994.

Jiasen Lu, Anitha Kannan, Jianwei Yang, Devi Parikh, and Dhruv Batra. 2017. Best of both worlds: Transferring knowledge from discriminative learning to a generative visual dialog model. In *Advances in Neural Information Processing Systems*, pages 314–324.

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.

Todd K Moon. 1996. The expectation-maximization algorithm. *IEEE Signal processing magazine*, 13(6):47–60.

Vishvak Murahari, Dhruv Batra, Devi Parikh, and Abhishek Das. 2020. Large-scale pretraining for visual dialog: A simple state-of-the-art baseline. *Proceedings of the European Conference on Computer Vision*.

Van-Quang Nguyen, Masanori Suganuma, and Takayuki Okatani. 2020. Efficient attention mechanism for visual dialog that can handle all the interactions between multiple inputs. *Proceedings of the European Conference on Computer Vision*.

Yulei Niu, Hanwang Zhang, Manli Zhang, Jianhong Zhang, Zhiwu Lu, and Ji-Rong Wen. 2019. Recursive visual attention in visual dialog. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6679–6688.

Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. 2017. Automatic differentiation in pytorch.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 1532–1543.

Jiaxin Qi, Yulei Niu, Jianqiang Huang, and Hanwang Zhang. 2020. Two causal principles for improving visual dialog. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.

Mengye Ren, Ryan Kiros, and Richard Zemel. 2015a. Exploring models and data for image question answering. In *Advances in Neural Information Processing Systems*, pages 2953–2961.

Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. 2015b. Faster R-CNN: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems*, pages 91–99.

Idan Schwartz, Seunghak Yu, Tamir Hazan, and Alexander G Schwing. 2019. Factor graph attention.

In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2039–2048.

Damien Teney, Lingqiao Liu, and Anton van Den Hengel. 2017. Graph-structured representations for visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. 2017. Graph attention networks. *arXiv preprint arXiv:1710.10903*.

Peng Wang, Qi Wu, Jiewei Cao, Chunhua Shen, Lianli Gao, and Anton van den Hengel. 2019. Neighbourhood watch: Referring expression comprehension via language-guided graph attention networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1960–1968.

Yue Wang, Shafiq Joty, Michael R Lyu, Irwin King, Caiming Xiong, and Steven CH Hoi. 2020. VD-BERT: A unified vision and dialog transformer with bert. *arXiv preprint arXiv:2004.13278*.

Qi Wu, Peng Wang, Chunhua Shen, Ian Reid, and Anton van den Hengel. 2018. Are you talking to me? reasoned visual dialog generation through adversarial learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6106–6115.

Zhang Xinyi and Lihui Chen. 2018. Capsule graph neural network. In *International conference on learning representations*.

Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. 2015. Show, attend and tell: Neural image caption generation with visual attention. In *Proceedings of International Conference on Machine Learning*, pages 2048–2057.

Ting Yao, Yingwei Pan, Yehao Li, and Tao Mei. 2018. Exploring visual relationship for image captioning. In *Proceedings of the European conference on computer vision (ECCV)*, pages 684–699.

Chuxu Zhang, Dongjin Song, Chao Huang, Ananthram Swami, and Nitesh V Chawla. 2019a. Heterogeneous graph neural network. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 793–803.

Ji Zhang, Yannis Kalantidis, Marcus Rohrbach, Manohar Paluri, Ahmed Elgammal, and Mohamed Elhoseiny. 2019b. Large-scale visual relationship

understanding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 9185–9194.

Zilong Zheng, Wenguan Wang, Siyuan Qi, and Song-Chun Zhu. 2019. Reasoning visual dialogs with structural and partial observations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6669–6678.

A Relation-aware Graph-over-Graph Network

A.1 Question Graph Construction

In practice, we observe that two words in a sentence usually hold certain relations. Such relations can be identified by the Neural Dependency Parsing (Dozat and Manning, 2017). As shown in Table 6, we list a part of commonly-used dependency relations.

A.2 Attention Alignment Module

After obtaining relation-aware features, we fuse the question representation q^*, history representation h^*, visual representation v^* through a multi-modal fusion strategy. We can use any existing multi-modal fusion method to learn a joint representation $\mathcal{J}$:

$$\mathcal{J} = \mathcal{F}(q^*, h^*, v^*; \Theta), \qquad (12)$$

where $\mathcal{J}$ is a multi-modal fusion method and Θ are trainable parameters of the fusion module. Here we utilize an efficient attention mechanism method (Nguyen et al., 2020) to fuse the multi-modal information, which is the state-of-the-art model in visual dialog.

Let $A_X(Y)$ denotes the efficient attention mechanism (Nguyen et al., 2020) from the information X to the information Y. For example, $A_{v^*}(v^*)$ denotes the efficient self-attention. The fused visual representation is obtained as follows:

$$v_{contact} = [A_{v^*}(v^*), A_{q^*}(v^*), A_{h^*}(v^*)], \quad (13)$$

$$\overline{v}' = \text{LayerNorm}(\sigma(v_{concat}W_{v^*}) + v^*) \quad (14)$$

$$a_V = \text{softmax}(W_{V_1}\sigma(W_{V_2}\overline{v}')), \qquad (15)$$

$$\overline{v} = \sum_{i=1}^{\mu} a_{V,i}\overline{v}'_i, \qquad (16)$$

where W_{v^*}, W_{V_1}, W_{V_2} are learned parameters. $\overline{q}$ and $\overline{h}$ can be obtained similarly. Thus, the joint representation $\mathcal{J}$ is obtained:

$$\mathcal{J} = W_J[\overline{q}, \overline{h}, \overline{v}], \qquad (17)$$

where W_J is a learned parameter.

Relations	Relation Description	Proportion
nsubj	nominal subject	16.1%
root	root node	16.0%
dep	dependent	15.7%
punct	punctuation	14.3%
det	determiner	9.0%
cop	copula	9.0%
prep	prepositional modifier	4.6%
aux	auxiliary	4.0%
pobj	object of a preposition	3.6%
amod	adjective modifier	3.2%
advmod	adverbial modifier	2.5%
dobj	direct object	1.5%

Table 6: The main categories of relations classified by the dependency parsing tool (Dozat and Manning, 2017) in VisDial v1.0 training split (Das et al., 2017).

A.3 Generative and Discriminative Decoders

Following Das et al. (Das et al., 2017), we consider both generative and discriminative decoders to score the candidate answers using the likelihood scores and the log-likelihood scores, respectively.

Generative Decoder Following Das et al. (Das et al., 2017), we design the generative decoder to score the candidate answers using the log-likelihood scores. Specifically, the generative decoder utilizes a two-layer LSTM (Hochreiter and Schmidhuber, 1997) to generate an answer using the context vector $\mathcal{J}$ as the initial hidden state. In the training phase, the generative decoder generates the next token based on the current token from the ground truth answer. In detail, we first append the special token "SOS" at the beginning of the ground truth answer and "EOS" at the end. We use Glove (Pennington et al., 2014) to initialize the embedding and obtain the embedding vectors $a_{gt} = [w_0, w_1, \ldots, w_N]$ where w_0 is the embedding of "SOS" and w_N is the embedding of "EOS". The hidden state h_n at timestep n is computed as follows:

$$h_n = \text{LSTM}(w_{n-1}, h_{n-1}), \qquad (18)$$

where h_0 is intializaed by $\mathcal{J}$. Then we obtain the log-likelihood of n-th word as follows:

$$p = \text{logsoftmax}(Wh + b), \qquad (19)$$

where W and b are learned parameters. In the training phase, we minimize the summation of the

Model	VisDial v0.9 (val)					VisDial v1.0 (val)					
	MRR↑	R@1↑	R@5↑	R@10↑	Mean↓	NDCG↑	MRR↑	R@1↑	R@5↑	R@10↑	Mean↓
Fusion-based Models											
LF (Das et al., 2017)	51.99	41.83	61.78	67.59	17.07	-	-	-	-	-	-
HRE (Das et al., 2017)	52.37	42.23	62.28	68.11	16.97	-	-	-	-	-	-
Attention-based Model											
MN (Das et al., 2017)	52.59	42.29	62.85	68.88	17.06	51.86	47.99	38.18	57.54	64.32	18.60
HCIAE (Lu et al., 2017)	53.86	44.06	63.55	69.24	16.01	59.70	49.07	39.72	58.23	64.73	18.43
CorefNMN (Kottur et al., 2018)	53.50	43.66	63.54	69.93	15.69	-	-	-	-	-	-
CoAtt (Wu et al., 2018)	54.11	44.32	63.82	69.75	16.47	59.24	49.64	40.09	59.37	65.92	17.86
RvA (Niu et al., 2019)	55.43	45.37	65.27	72.97	10.71	-	-	-	-	-	-
DVAN (Guo et al., 2019b)	55.94	46.58	65.50	71.25	14.79	-	-	-	-	-	-
Primary (Guo et al., 2019a)	-	-	-	-	-	-	49.01	38.54	59.82	66.94	16.60
ReDAN (Gan et al., 2019)	-	-	-	-	-	60.47	50.02	40.27	59.93	66.78	17.40
DMRM (Chen et al., 2020a)	55.96	46.20	66.02	72.43	13.15	-	50.16	40.15	60.02	67.21	15.19
DAM (Jiang et al., 2020c)	-	-	-	-	-	60.93	50.51	40.53	60.84	67.94	16.65
Pretraining-based Model											
VDBERT (Wang et al., 2020)	55.95	46.83	65.43	72.05	13.18	-	-	-	-	-	-
Graph-based Model											
KBGN (Jiang et al., 2020a)	-	-	-	-	-	60.42	50.05	40.40	60.11	66.82	17.54
LTMI (Nguyen et al., 2020)[†]	55.85	46.07	65.97	72.44	14.17	61.61	50.38	40.30	60.72	68.44	15.73
GoG-Gen (Ours)	**56.32**	**46.65**	**66.41**	72.69	13.78	**62.63**	**51.32**	**41.25**	**61.83**	**69.44**	**15.32**
GoG-Multi-Gen (Ours)	56.89	47.04	66.92	72.87	13.45	63.35	51.80	41.78	62.23	69.79	15.16
GoG-Multi (Ours)	59.38	48.58	71.33	78.78	9.94	65.20	55.38	43.93	68.22	76.75	9.98

Table 7: Main comparisons on both VisDial v0.9 and v1.0 datasets using the generative decoder. † denotes that we re-implemented the model. Underline indicates the highest performance among previous approaches except pretraining-base models.

negative log-likelihood $\mathcal{L}_{\mathcal{G}}$ defined by:

$$\mathcal{L}_{\mathcal{G}} = -\sum_{n=1}^{N} p_n. \qquad (20)$$

In the validation and test phase, we compute the summation s_i of the log-likelihood for each candidate answer $\hat{a}_i$:

$$s_i = \sum_{n=1}^{N} p_n^{\hat{a}_i}. \qquad (21)$$

Then, the rankings of the candidate answers are derived as $\mathrm{softmax}(s_1, \ldots, s_{100})$.

Discriminative Decoder A discriminative decoder outputs the likelihood score for each of 100 candidate answers for the current question. Similar to the generative decoder, we use LSTM to obtain the hidden state h_n for b-th word but we do not use context vector J to initialize the h_0. The representation of each candidate answer is $a_i = h_N$. The score p_i for i-th candidate answer is computed by:

$$p = \mathrm{logsoftmax}(a_1^T \mathcal{J}, \ldots, a_{100}^T \mathcal{J}) \qquad (22)$$

In the test phase, we sort the candidate answers using these scores. In the training phase, the cross-entropy loss $\mathcal{L}_{\mathcal{D}}$.

A.4 Multi-Task Learning

According to (Nguyen et al., 2020), we apply our GoG to the state-of-the-art model (Nguyen et al., 2020) in the multi-task learning setting that accuracy is improved by training the entire network using the two decoders simultaneously. This is simply done by minimizing the sum of the losses, $\mathcal{L}_{\mathcal{D}}$ for the discriminative one and $\mathcal{L}_{\mathcal{G}}$ for the generative one:

$$\mathcal{L} = \mathcal{L}_{\mathcal{D}} + \mathcal{L}_{\mathcal{G}} \qquad (23)$$

The increase in performance may be attributable to the synergy of learning two tasks while sharing the same encoder.

## B Experiments

B.1 Main Results

Comparison with previous approaches using generative decoders. As shown in Table 7, we provide the full comparison with all the previous generative approaches.

B.2 Qualitative Results

More examples generated and retrieved by our GoG are provided in Figure 7. Due to the limited number of pages, we only provide an additional example of Figure 7.
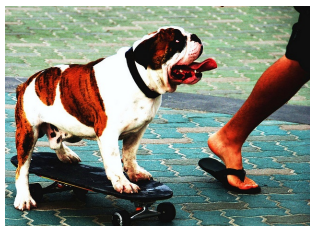
chocolate cake with fresh strawberries and small nuts
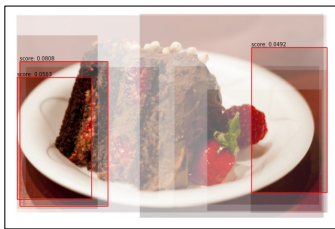
Q: is it on a table ?
GT: i think it
Gen: yes think so
Disc yes

Q: is there any silverware ?
GT: no
Gen: no
Disc no

a brown and white dog riding a skateboard

Q: can you see any people ?
GT: part of a person
Gen: just of 1 person
Disc part of a person

Q: what color is the skateboard ?
GT: black
Gen: black
Disc black

a man holding a disk smiles as he leans against a sign at a disc golf course

Q: how many discs are shown ?
GT: 1
Gen: 1
Disc 3

Q: what color disk ?
GT: yellow
Gen: yellow
Disc yellow

a man playing tennis is attempting to hit the ball with his racket

Q: how old is the man ?
GT: i don ' t know probably in late 30s
Gen: maybe would ' t know
Disc looks middle aged

Q: does he wear sunglasses ?
GT: yes
Gen: yes
Disc yes

Figure 7: Examples of dialogs generated and retrieved by our model. blue denotes has the same meaning with the ground truth and red denotes wrong answers.