Learning to Ground Visual Objects for Visual Dialog

Feilong Chen  Xiuyi Chen  Can Xu  Daxin Jiang*
Microsoft Corporation, Beijing, China
{ivess.chan, hugheren.chan}@gmail.com
{caxu, djiang}@microsoft.com

Abstract

Visual dialog is challenging since it needs to answer a series of coherent questions based on understanding the visual environment. How to ground related visual objects is one of the key problems. Previous studies utilize the question and history to attend to the image and achieve satisfactory performance, however these methods are not sufficient to locate related visual objects without any guidance. The inappropriate grounding of visual objects prohibits the performance of visual dialog models. In this paper, we propose a novel approach to Learn to Ground visual objects for visual dialog, which employs a novel visual objects grounding mechanism where both prior and posterior distributions over visual objects are used to facilitate visual objects grounding. Specifically, a posterior distribution over visual objects is inferred from both context (history and questions) and answers, and it ensures the appropriate grounding of visual objects during the training process. Meanwhile, a prior distribution, which is inferred from context only, is used to approximate the posterior distribution so that appropriate visual objects can be grounded even without answers during the inference process. Experimental results on the VisDial v0.9 and v1.0 datasets demonstrate that our approach improves the previous strong models in both generative and discriminative settings by a significant margin.

1 Introduction

With the development of deep learning, various vision-language tasks have been introduced and attracted widespread attention, 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; Wu et al., 2018; Kottur et al., 2018; Gan et al., 2019; Guo et al., 2019b), and visual dialog (Das et al., 2017; Chen et al., 2021a; Agarwal et al., 2020; Chen et al., 2021b; Qi et al., 2020). Specifically, visual dialog, which aims to hold a meaningful conversation (Chen et al., 2021c, 2020b) with a human about a given image, is a challenging task that requires models to locate related visual objects in an image and answer the current question based on the history and the located visual objects.

In order to answer the question correctly, we need to accurately locate the question-related visual objects. Most existing methods utilize kinds of attention mechanism (Lu et al., 2017; Wu et al., 2018; Kottur et al., 2018; Gan et al., 2019; Guo et al., 2019b) to capture the target visual objects. ReDAN (Gan et al., 2019) and DMAM (Chen et al., 2020a) use multi-step reasoning based on dual attention to iteratively update related visual objects. 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 and obtain weight distributions. Moreover, there are some approaches (Zheng et al., 2019; Schwartz et al., 2019; Jiang et al., 2020b; Guo et al., 2020; Jiang et al., 2020a) using graph-
based structures to capture related visual objects. 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 to ground visual objects. CAG (Guo et al., 2020) focuses on an iterative question-conditioned context-aware graph to locate related visual objects. However, the methods mentioned above obtain the prior distribution of visual objects through various interactions of questions, history and images, and finally use the prior distribution to obtain the final representation of the image. The prior distribution of visual objects is not enough to ground accurate visual objects, thus obtaining the wrong representation of the image.

In this paper, we propose a method to learn to ground visual objects in visual dialog. Specifically, we obtain the posterior distribution over visual objects by utilizing contexts and answers, while the prior distribution works without knowing answers in advance. Then we minimize the distance between the two distributions. During the training process, our model is trained to minimize the KL divergence between the prior distribution and the posterior distribution so that our model can approximate the posterior distribution accurately using the prior distribution. Then, during the inference process, the model grounds visual objects merely based on the prior distribution (i.e., without any posterior information). We show that through this process, the model can effectively learn to ground visual objects accurately and give informative and accurate responses by utilizing appropriate visual objects. We test the effectiveness of our proposed model on two large-scale datasets: VisDial v0.9 and v1.0 (Das et al., 2017). The contributions of this work are summarized as follows:

- We explore the importance of answers in grounding visual objects related to questions in visual dialog.
- We propose a novel approach to realize learning to ground visual objects in visual dialog via bridging the gap between the prior and posterior distribution over visual objects.
- We conduct extensive experiments and ablation studies on two large-scale datasets VisDial v0.9 and v1.0. Experimental results show that our approach can be used to improve previous visual dialog models in both generative and discriminative settings.

2 Methodology

Following Das et al. (2017), a visual dialog agent is given three inputs, i.e., an image \( i \), history \( h \) (the caption and question-answer pairs) till round \( t−1 \): 

\[
\begin{align*}
    h &= (Cap, (q_1, a_1), \ldots, (q_{t−1}, a_{t−1})) \\
    h_0, h_1, \ldots, h_{t−1}
\end{align*}
\]

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 an answer \( a_t \) to the question \( q_t \).

2.1 Model Architecture

In this paper, we focus on training a neural visual dialog model with an effective visual objects grounding mechanism. As shown in Figure 2, we simplify existing visual dialog models into five major components:

- **The context encoder** encodes dialog history \( h \) and the current question \( q_t \) with an attention mechanism into a context vector \( x_c \), and feeds it into the visual objects manager and the decoder.

- **The visual encoder** takes the image \( i \) as input and extract the image features \( v = \{v_1, v_2, \ldots, v_{\mu}\} \) where \( \mu \) denotes the number of object proposals for each image. Each object proposal is represented by a \( dv \)-dimension feature vector.

- **The answer encoder** encodes the ground-truth answer \( a_t \) into a response vector \( y_r \), and feeds it into the visual objects manager.

- **The visual object manager** consists of two sub-modules: a prior module and a posterior module.
module. Given the previously encoded $x$ and $v^\mu_{i=1}$ (and $y$ if available), the visual object manager is responsible for deciding an appropriate distribution over visual objects and feeds the weighted visual object features $v_s$ (together with an attention-based context vector $x$) into the decoder.

- The decoder generates and retrieves responses based on the visual object feature $v_s$ and the attention-based context vector $x$.

### 2.2 Our Approach

When given the context vector $x$ and the visual object features $v = \{v_1, v_2, \ldots, v_\mu\}$, and response vector $y$, the goal of the visual object manager is to decide an appropriate distribution $D$ over visual objects and obtain the weighted visual object representation $v_s$ based on the distribution $D$.

The visual object manager consists of two sub-modules: a prior module and a posterior module.

#### The Prior Module

The prior module aims to calculate the conditional probability distribution over $\mu$ visual objects, denoted by $p(v|x)$:

$$p(v = v_i|x) = \frac{\exp(f_{cv}(x, v_i))}{\sum_{j=1}^{\mu} \exp(f_{cv}(x, v_j))},$$

where $f_{cv}(\cdot, \cdot)$ denotes the interaction function of the context vector $x$ and the visual object features $v_i$. For example, $f_{cv}(\cdot, \cdot)$ can be the dot product, self-attention or other mechanisms to measure the association between $v_i$ and the context vector $x$. A high association means that $v_i$ is relevant to $x$ and thus, $v_i$ has a larger weight. Note that $p(v|x)$ is conditioned only on $x$ and thus, it is a prior distribution over visual objects since it works without knowing the response. However, there can be different visual objects that are relevant to the contexts, and thus, it is difficult to select visual objects simply based on the prior distribution in training.

#### The Posterior Module

Motivated by this, in the posterior module, we define a posterior distribution over visual objects, denoted by $p(v|x, y)$, by considering both contexts and responses:

$$p(v = v_i|x, y) = \frac{\exp(f_{cv}(f_{cy}(x, y), v_i))}{\sum_{j=1}^{\mu} \exp(f_{cv}(f_{cy}(x, y), v_j))},$$

where $f_{cy}(\cdot, \cdot)$ denotes the interaction function of $x$ and $y$. For example, the $f_{cy}(\cdot, \cdot)$ can be an add operation, fully connected layer and other methods. Compared with the prior distribution, the posterior distribution is sharp since the actual visual objects used in the true response $a_t$ can be captured.

#### Bridging the Gap

Clearly, the discrepancy between prior and posterior distributions introduces great challenges in training the model: it is desirable to ground visual objects based on the posterior
distribution, which, however, is unknown during inference. In this paper, we propose to approximate the posterior distribution using the prior distribution so that our model is capable of selecting appropriate visual objects even without posterior information. For this purpose, we introduce an auxiliary loss, namely the Kullback-Leibler divergence loss (KLDivLoss), to bridge the gap between the prior distribution and the posterior distribution. The KLDivLoss is defined as follows:

$$\mathcal{L}_{KL} = \sum_{i=1}^{\mu} p(v = v_i|x,y) \log \left( \frac{p(v = v_i|x,y)}{p(v = v_i|x)} \right).$$  \hspace{1cm} (3)

When minimizing KLDivLoss, the posterior distribution $p(v|x,y)$ can be regarded as labels and our model is instructed to use the prior distribution $p(v|x)$ to approximate $p(v|x,y)$ accurately. As a consequence, even when the posterior distribution is unknown in the inference process (since the actual response $a_t$ is unknown), the prior distribution $p(v|x)$ can be effectively utilized to ground appropriate visual objects so as to generate and retrieve proper responses. To the best of our knowledge, it is the first neural model in visual dialog, which incorporates the posterior distribution as guidance, enabling accurate visual object grounding and high-quality response generation and retrieval.

3 Application of Our Approach

We take the strong baseline LTMI (Nguyen et al., 2020) as a base model to introduce our approach, which mainly consists of the following components:

**Context Encoder and Answer Encoder:** We use two bi-directional LSTM encoders to extract token-level representations $Q \in \mathbb{R}^{\lambda \times d_q}$ and $y \in \mathbb{R}^{\lambda \times d_q}$ of the question $q_t$ and the answer $a_t$. We use another bi-directional LSTM encoder to extract sentence-level representations $H \in \mathbb{R}^{T \times d_q}$ of the history $h$. $\lambda$ is the length of questions and answers with paddings, $T$ is the turn of dialog and $d_q$ is the dimension. $Q$ and $H$ are fused into a context representation $x$ with multi-head attention (Vaswani et al., 2017).

**Visual Encoder:** Similar to (Anderson et al., 2018), we extract the image features by using a pre-trained Faster RCNN (Ren et al., 2015b). We select $\mu$ object proposals for each image, where each object proposal is represented by a 2048-dimension feature vector. We transform the obtained visual region features by a multi-layer perception and obtain the image features $I = I_i^\mu_{j=0} \in \mathbb{R}^{\mu \times d_q}$.

**Prior Module:** We use multi-head attention (Vaswani et al., 2017) as $f_{cv}(\cdot, \cdot)$ to manage the multi-modal interaction. A cross-attention layer is firstly applied to outputs of the textual and visual encoders:

$$P = \text{softmax}(I x^T) \in \mathbb{R}^{\mu \times \lambda}, \hspace{1cm} (4)$$
$$I_x = \text{CrossAttn}(I, x) = P x \in \mathbb{R}^{\mu \times d_q}, \hspace{1cm} (5)$$

where the softmax conducts the normalization over each column of the matrix. We convert the representation $\hat{I}$ into $d_q$-dimension vectors $V$. This conversion is performed by a simple self-attention computation as follows:

$$g = \text{softmax}(\text{ReLU}(\hat{I} W_1 + b_1) W_2 + b_2), \hspace{1cm} (6)$$

where $g \in \mathbb{R}^{\mu \times 1}$, $W_1$, $W_2$, $b_1$, $b_2$ are learned parameters. We obtain the representation $V$ as follows:

$$v_{prior} = g^T \hat{I} \in \mathbb{R}^{d_q}. \hspace{1cm} (7)$$

$g$ is regarded as the prior distribution over visual objects.

**Posterior Module:** We simply utilize the add operation as $f_{cy}(\cdot, \cdot)$ to manage the interaction of $x$ and $y$:

$$x_y = x + y \hspace{1cm} (8)$$

We replace $x$ in Eq.(6) - Eq.(8) with $x_y$ and thus obtain the posterior distribution $G$ and $v_{post}$.

**Generative and Discriminative Decoder:** We utilize another LSTM as our discriminative and generative decoders following the previous studies (Das et al., 2017; Nguyen et al., 2020). Receiving the representation of context, images and the candidate answers, the two decoders compute the score of each candidate answer in different ways. The objective function of the base model is to minimize the negative log-likelihood $\mathcal{L}_G$ of answer generated for the generative decoder or the cross-entropy loss $\mathcal{L}_D$ for the discriminative decoder. We utilize the Kullback-Leibler (KL) divergence loss to narrow the gap. The objective functions of the student are as follows:

$$\mathcal{L} = \mathcal{L}_G + \lambda \mathcal{L}_{KL}(G,g), \hspace{1cm} (9)$$
$$\mathcal{L} = \mathcal{L}_D + \lambda \mathcal{L}_{KL}(G,g), \hspace{1cm} (10)$$
$$\mathcal{L} = \mathcal{L}_G + \mathcal{L}_D + \lambda \mathcal{L}_{KL}(G,g), \hspace{1cm} (11)$$
4 Experiments

4.1 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 \( \mu = 100 \) objects from each image. Our model is implemented based on PyTorch (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 with a learning rate of 0.001, 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 hours / 1 epoch for the discriminative setting and 1 hour / 1 epoch for the generative setting. Our student model is the same as LTMI, with the total parameters 42.20M.

The \( \lambda \) sets to 1.

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, apart from the image, ground-truth dialog history and the question, 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 is human-generated (M1). We record the percentage of responses that are evaluated better than or equal to the last response in the dialog (M2), according to the human subjects’ evaluation.

4.2 Main Results

Baseline methods. In our experiment, compared methods can be grouped into four types: (1) Fusion-based models: LF (Das et al., 2017) and HREA (Das et al., 2017). (2) Attention-based models: HCIAE (Lu et al., 2017), CoAtt (Wu et al., 2018), Primary (Guo et al., 2019a), ReDAN (Gan et al., 2019), CorefNMN (Kottur et al., 2018), ReDAN (Gan et al., 2019), CorefNMN (Kottur et al., 2018),
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 which is unfair compared with other models. Underline indicates the highest performance among previous approaches except for the pretraining-based models. Our approach improves the strong baseline significantly. (t-test, p-value < 0.01)

<table>
<thead>
<tr>
<th>Model</th>
<th>VisDial v0.9 (val)</th>
<th>VisDial v1.0 (test-std)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>MRR ↑</td>
<td>R@1 ↑</td>
</tr>
<tr>
<td>ReDAN (Gan et al., 2019)</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>MCA (Agarwal et al., 2020)</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>GNN-EM (Zheng et al., 2019)</td>
<td>62.85</td>
<td>48.95</td>
</tr>
<tr>
<td>DualVD (Jiang et al., 2020)</td>
<td>62.94</td>
<td>48.64</td>
</tr>
<tr>
<td>PGA (Schwartz et al., 2019)</td>
<td>65.25</td>
<td>51.43</td>
</tr>
<tr>
<td>CAG (Guo et al., 2020)</td>
<td>67.56</td>
<td>54.64</td>
</tr>
<tr>
<td>KBGN (Jiang et al., 2020a)</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>VisualBERT (Murahari et al., 2020)†</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>VDBERT (Wang et al., 2020) ‡</td>
<td>70.04</td>
<td>57.79</td>
</tr>
<tr>
<td>LTMI (Nguyen et al., 2020)</td>
<td>66.41</td>
<td>53.36</td>
</tr>
<tr>
<td>LTMI-LG (Ours)</td>
<td>67.63</td>
<td>54.69</td>
</tr>
</tbody>
</table>

Table 3: Accuracy of visual grounding with and without knowing the answer. We randomly sample 1000 samples and ask human annotators to ground the three most likely objects from the image.

<table>
<thead>
<tr>
<th>Model</th>
<th>w/o Ans</th>
<th>w/ Ans</th>
<th>with LG</th>
</tr>
</thead>
<tbody>
<tr>
<td>LTMI (Nguyen et al., 2020)</td>
<td>68.6</td>
<td>97.1</td>
<td>82.1</td>
</tr>
<tr>
<td>Random</td>
<td>3.6</td>
<td>3.6</td>
<td>-</td>
</tr>
<tr>
<td>Human</td>
<td>96.7</td>
<td>99.3</td>
<td>-</td>
</tr>
</tbody>
</table>

Table 4: Effects of different visual objects distribution.

<table>
<thead>
<tr>
<th>Model</th>
<th>NDCG</th>
<th>MRR</th>
<th>R@1</th>
<th>R@5</th>
<th>R@10</th>
<th>Mean</th>
</tr>
</thead>
<tbody>
<tr>
<td>LTMI</td>
<td>61.61</td>
<td>50.38</td>
<td>40.30</td>
<td>60.72</td>
<td>68.44</td>
<td>15.73</td>
</tr>
<tr>
<td>LTMI-Mean</td>
<td>56.66</td>
<td>43.64</td>
<td>32.59</td>
<td>54.66</td>
<td>62.91</td>
<td>17.59</td>
</tr>
<tr>
<td>LTMI-Random</td>
<td>56.89</td>
<td>43.70</td>
<td>33.01</td>
<td>54.76</td>
<td>62.76</td>
<td>17.76</td>
</tr>
<tr>
<td>LTMI-LG</td>
<td>63.23</td>
<td>51.30</td>
<td>41.34</td>
<td>61.61</td>
<td>69.06</td>
<td>15.26</td>
</tr>
<tr>
<td>LTMI-Human</td>
<td>70.10</td>
<td>63.96</td>
<td>50.74</td>
<td>69.29</td>
<td>80.02</td>
<td>8.12</td>
</tr>
</tbody>
</table>

Table 5: Main comparisons on VisDial v1.0 val datasets using the discriminative decoder. ◦ denotes that the model utilizes large extra datasets for training. † denotes that the model trains from scratch.

<table>
<thead>
<tr>
<th>Model</th>
<th>NDCG</th>
<th>MRR</th>
<th>R@1</th>
<th>R@5</th>
<th>R@10</th>
<th>Mean</th>
</tr>
</thead>
<tbody>
<tr>
<td>MN (Das et al., 2017)</td>
<td>-</td>
<td>60.29</td>
<td>46.14</td>
<td>77.68</td>
<td>87.57</td>
<td>4.84</td>
</tr>
<tr>
<td>HCIAE (Li et al., 2017)</td>
<td>-</td>
<td>61.96</td>
<td>48.25</td>
<td>78.97</td>
<td>88.43</td>
<td>4.56</td>
</tr>
<tr>
<td>CoAtt (Wu et al., 2018)</td>
<td>-</td>
<td>62.77</td>
<td>49.38</td>
<td>78.99</td>
<td>88.49</td>
<td>4.56</td>
</tr>
<tr>
<td>ReDAN (Gan et al., 2019)</td>
<td>-</td>
<td>64.29</td>
<td>50.65</td>
<td>81.29</td>
<td>90.57</td>
<td>4.10</td>
</tr>
<tr>
<td>KBGN (Jiang et al., 2020a)</td>
<td>59.08</td>
<td>64.86</td>
<td>51.37</td>
<td>81.71</td>
<td>90.54</td>
<td>4.00</td>
</tr>
<tr>
<td>VDBERT (Wang et al., 2020)</td>
<td>56.20</td>
<td>62.25</td>
<td>48.16</td>
<td>79.57</td>
<td>89.01</td>
<td>4.31</td>
</tr>
<tr>
<td>VDBERT (Wang et al., 2020) ‡</td>
<td>63.22</td>
<td>67.44</td>
<td>54.02</td>
<td>83.96</td>
<td>92.33</td>
<td>3.53</td>
</tr>
<tr>
<td>LTMI (Nguyen et al., 2020)</td>
<td>62.72</td>
<td>62.32</td>
<td>48.94</td>
<td>78.65</td>
<td>87.88</td>
<td>4.86</td>
</tr>
<tr>
<td>LTMI-LG (Ours)</td>
<td>59.67</td>
<td>63.03</td>
<td>51.69</td>
<td>81.49</td>
<td>90.32</td>
<td>4.02</td>
</tr>
</tbody>
</table>

Table 6: Main comparisons on VisDial v1.0 val datasets using the discriminative decoder. ◦ denotes that the model utilizes large extra datasets for training. † denotes that the model trains from scratch.
ent methods on the VisDial v1.0 val and VisDial v0.9 val. With the guidance of the teacher, we train our LTMI-LG with the ability of accurately grounding visual objects. As a result, our approach improve significantly (nearly 1% on all metrics) compared with LTMI (Nguyen et al., 2020). Comparing with the state-of-the-art results of different metrics, our model improves NDCG for 61.51 to 63.53 (+1.92), MRR from 50.51 to 51.43 (+0.92), R@1 from 40.53 to 41.68 (+1.13), R@5 from 60.84 to 61.96 (+1.12), R@10 from 68.44 to 69.87 (+1.43), Mean from 15.73 to 14.89 (+0.84) on the VisDial v1.0 val. Our model also brings a large improvement to LTMI (Nguyen et al., 2020) on the VisDial v0.9 val. The performance of our model exceeds the performance of VDBERT (Wang et al., 2020) on all the metrics except Mean. We believe data is an important factor in deep learning (LeCun et al., 2015). VDBERT (Wang et al., 2020) works because it uses a lot of extra data for training. The reason why our method is effective is that we use the teacher to teach the student visual grounding, which can be regarded as a kind of data annotation.

**Discriminative Results.** As shown in Table 2 and Table 5, we compare our method with previous works on the VisDial v1.0 test, VisDial v0.9 val and VisDial v1.0 val. Our model improves significantly compared with LTMI (Nguyen et al., 2020), improving about +3% on MRR, R@1, R@5 and R@10 on the VisDial v1.0 test. Our approach also brings a large improvement on the VisDial v0.9 and achieves the best results on MRR, R@1 and R@5 among non-pre-trained models. In a discriminative setting, our approach performs worse than pre-training models VisualBERT (Murahari et al., 2020) and VDBERT (Wang et al., 2020) because pre-training models utilize extra large-scale datasets to train the models which are unfair compared with other models. As shown in Table 5, VDBERT † which trains from scratch performs worse than our LTMI-LG.

### 4.3 Ablation Study

In order to transfer knowledge, we need a metric loss to measure the gap between teachers and students. In our main experiments, we utilize the kullback-leibler (KL) divergence loss to diminish the gap of the weight distribution between the student model and the teacher model, the mean squared loss to diminish the gap of the representation of images. To compare different losses, we utilize the mean squared loss for attention maps and KL loss for the representation of images as shown in Table 6. We find that KL loss is more suitable for attention distribution (better for NDCG, MRR and

<table>
<thead>
<tr>
<th>Model</th>
<th>NDCG</th>
<th>MRR</th>
<th>R@1</th>
<th>R@5</th>
<th>R@10</th>
<th>Mean</th>
</tr>
</thead>
<tbody>
<tr>
<td>LTMI†</td>
<td>61.61</td>
<td>50.38</td>
<td>40.30</td>
<td>60.72</td>
<td>68.44</td>
<td>15.73</td>
</tr>
<tr>
<td>LG-Attn-MSE</td>
<td>63.03</td>
<td>51.14</td>
<td>40.91</td>
<td>61.78</td>
<td>69.43</td>
<td>15.08</td>
</tr>
<tr>
<td>LG-Image-MSE</td>
<td>62.80</td>
<td>51.21</td>
<td>41.01</td>
<td>62.02</td>
<td>69.90</td>
<td>14.90</td>
</tr>
<tr>
<td>LG-Attn-KL</td>
<td>63.23</td>
<td>51.30</td>
<td>41.34</td>
<td>61.61</td>
<td>69.06</td>
<td>15.26</td>
</tr>
<tr>
<td>LG-Image-KL</td>
<td>62.36</td>
<td>51.24</td>
<td>41.19</td>
<td>61.73</td>
<td>69.31</td>
<td>15.13</td>
</tr>
<tr>
<td>Att-Image-MSE</td>
<td>62.73</td>
<td>51.19</td>
<td>40.97</td>
<td>61.80</td>
<td>69.43</td>
<td>15.10</td>
</tr>
</tbody>
</table>

Table 6: Ablation study on VisDial v1.0 val datasets using the generative decoder.
R@1) and MSE loss for the representation (better for R@5, R@10 and Mean). In addition, we use the attention via KL loss and the representation via MSE loss for distillation at the same time. The result is not so satisfactory and we think these two methods have some redundancy.

### 4.4 Case Study

As shown in Figure 4, we visualize the learned attention maps to understand the model. The colorful region means higher attention weights. We draw the bounding boxes of the first three highest scores. As shown in the top image in Figure 4, the question “is he standing ?” indicates the man’s overall posture rather than the local. LTMI grounds the wrong visual objects while our model grounds the right objects. As shown in the bottom image in Figure 4, the question “is his tennis racket up in the air swinging ?” concerns the racket rather than the tennis balls. Our model grounds accurately while LTMI makes mistakes. These examples show that our LTMI-LG has learned the ability to ground visual objects via our answer-aware knowledge distillation.

### 4.5 Human Study

As shown in Table 7, we conduct human study to further demonstrate the effectiveness of our model. Our model achieves the highest scores both on the metric M1 and M2 compared with LTMI.

### 5 Related Work

Recent several works (Shuster et al., 2018; Liang et al., 2021; Yang et al., 2020) explore leveraging visual information to enhance dialogue models. While visual dialog models focus on the intersection of questions, history and images. How to locate the related visual objects is quite important. MN (Das et al., 2017), HCIAE (Lu et al., 2017), CorefNMN (Kottur et al., 2018), CoAtt (Wu et al., 2018), RvA (Niu et al., 2019), DVAN (Guo et al., 2019b) utilize kinds of attention mechanisms as the backbone to locate the related visual objects. VisualBERT (Murahari et al., 2020) and VDBERT (Wang et al., 2020) exploit large extra datasets to explore in visual dialog via pretraining language models. GNN-EM (Zheng et al., 2019), FGA (Schwartz et al., 2019), DualVD (Jiang et al., 2020b), CAG (Guo et al., 2020) and KBGN (Jiang et al., 2020a) utilize graph neural networks to obtain the representation of visual objects. However, most existing visual dialog models condition visual objects simply on history and questions, which we regard as a prior distribution over visual objects. In this paper, we propose an approach to learn to ground visual objects via bridge the gap between the prior distribution and the posterior distribution.
6 Conclusion

In this paper, we propose a novel approach to learn to ground visual objects for visual dialog, which employs a novel visual objects grounding mechanism where both prior and posterior distributions over visual objects are used to facilitate visual objects grounding. Experimental results on two large-scale datasets show that our approach improves the previous models by a significant margin.

References


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 EMNLP.


Xiaoze Jiang, Siyi Du, Zengchang Qin, Yajing Sun, and Jing Yu. 2020a. KBGN: Knowledge-bridge graph network for adaptive vision-text reasoning in


