

# Enhancing Visual Dialog Questioner with Entity-based Strategy Learning and Augmented Guesser

Duo Zheng<sup>1\*</sup>, Zipeng Xu<sup>1\*</sup>, Fandong Meng<sup>2</sup>, Xiaojie Wang<sup>1†</sup>,  
Jiaan Wang<sup>3</sup>, Jie Zhou<sup>2</sup>

<sup>1</sup>Beijing University of Posts and Telecommunications

<sup>2</sup>Pattern Recognition Center, WeChat AI, Tencent, China

<sup>3</sup>Soochow University, Suzhou, China

{zd, xuzp, xjwang}@bupt.edu.cn, jawang1@stu.suda.edu.cn

{fandongmeng, withtomzhou}@tencent.com

## Abstract

Considering the importance of building a good Visual Dialog (VD) Questioner, many researchers study the topic under a Q-Bot-A-Bot image-guessing game setting, where the Questioner needs to raise a series of questions to collect information of an undisclosed image. Despite progress has been made in Supervised Learning (SL) and Reinforcement Learning (RL), issues still exist. Firstly, previous methods do not provide explicit and effective guidance for Questioner to generate visually related and informative questions. Secondly, the effect of RL is hampered by an incompetent component, i.e., the Guesser, who makes image predictions based on the generated dialogs and assigns rewards accordingly. To enhance VD Questioner: 1) we propose a **Related entity enhanced Questioner (ReeQ)** that generates questions under the guidance of related entities and learns entity-based questioning strategy from human dialogs; 2) we propose an **Augmented Guesser (AugG)** that is strong and is optimized for the VD setting especially. Experimental results on the VisDial v1.0 dataset show that our approach achieves state-of-the-art performance on both image-guessing task and question diversity. Human study further proves that our model generates more visually related, informative and coherent questions.

## 1 Introduction

Visual Dialog (VD), which expects AI agents to conduct visually related dialog, has attracted growing interests due to its research significance and application prospects. Most of the work (Lu et al., 2017; Niu et al., 2019; Gan et al., 2019; Chen et al., 2020; Agarwal et al., 2020; Nguyen et al., 2020; Chen et al., 2021) pays attention to modeling an Answerer agent. However, it is also important to

model a VD Questioner agent that can constantly ask visually related and informative questions.

Previous researches (Das et al., 2017b; Murahari et al., 2019a; Zhou et al., 2019) have explored building open-domain VD Questioner under a Q-Bot-A-Bot image-guessing game setting, namely GuessWhich (Das et al., 2017b). Given an undisclosed image, GuessWhich can be regarded to have two stages: 1) Dialog generation stage: Q-Bot (Questioner, who only knows a caption of the image at first) successively asks questions to collect information about the image, and A-Bot (Answerer, who can see the image) answers the questions. 2) Guess stage: Q-Bot guesses the target image based on the generated dialog. Corresponding to the two stages, Q-Bot has two roles, i.e. Question Generator (QGen) and Guesser<sup>1</sup>. Besides Supervised Learning (SL), previous methods (Das et al., 2017b; Murahari et al., 2019a; Zhou et al., 2019) introduce Reinforcement Learning (RL) to further fine-tune the agent. Though progress has been made, issues still exist.

Firstly, previous work does not provide explicit and effective guidance to generate visually related and informative questions. To encourage diverse questions, Murahari et al. (2019a) penalize the similarity in successive textual dialog hidden states. But this method can not promise the diverse questions are visually related. To ask visually related questions, Zhou et al. (2019) retrieve the most likely image at each round to provide Questioner with visual information. Yet, an image contains many contents while the method does not provide explicit guidance for Questioner to ask about which one.

Secondly, the reward in RL is not efficient due to an incompetent Guesser, hampering the effect of RL optimization. At each round of the dialog, Guesser makes an image feature prediction based

\* Equal contribution. Work was done when Zheng and Xu were interning at Pattern Recognition Center, WeChat AI, Tencent Inc, China.

† Xiaojie Wang is the corresponding author.

<sup>1</sup>We borrow the two concepts from GuessWhat?! (de Vries et al., 2017) to clarify the two correspondingly identical roles of Q-Bot in the GuessWhich setting.

$\alpha$	PMR $\uparrow$	Unique questions $\uparrow$
500	93.91	<b>6.39</b>
1000	96.22	6.22
2000	96.48	6.09
4000	<b>96.60</b>	4.36

Table 1: In previous method (Das et al., 2017b), a good QGen with higher unique questions, is together with a limited Guesser with lower PMR (Percentile Mean Rank).  $\alpha$  is the loss ratio of Guesser to QGen in cooperative training.<sup>2</sup>

on current dialog, then the reward is assigned to encourage the reduce of the distance between image feature prediction and target image feature. The efficiency of reward relies on the performance of Guesser heavily. However, previous Guessers’ performance is limited. This results from a cooperative training setting, where Guesser shares the same encoder with QGen and is optimized jointly. As illustrated in Tab. 1, using previous method, it is impossible to simultaneously obtain a good QGen and a good Guesser. Conventionally, since QGen’s performance is of higher priority to be concerned, the performance of Guesser consequently becomes inferior. As they use this limited Guesser to assign reward in RL, the reward is likely to be uncertain and thus inefficient. The effect of RL optimization is hampered consequently. Further progress requires a stronger Guesser to assign reliable rewards.

To remedy above issues, we propose a **Related entity enhanced Questioner (ReeQ)** and an **Augmented Guesser (AugG)** to enhance the VD Questioner in both SL and RL. ReeQ is a Questioner that explicitly uses related entities as guidance to generate questions and learns entity-based questioning strategy through large-scale human dialogs. In concrete, ReeQ firstly uses the image caption to retrieve related entities, which are pre-processed to be related to the target image; then at each round of the dialog, it selects which entity to ask about according to current dialog condition; lastly, it uses the selected entity as a hint to guide question generation. The related entities help ReeQ to ask visually-related questions while questioning strategy-learning enables it to ask constantly informative questions. AugG is a strong Guesser that is optimized with a special consideration for the VD setting. Specifically, we separately train

<sup>2</sup>As they find change of the loss ratio will lead to different results on PMR, we conduct experiments that train the model with different loss ratios using their code (Modhe et al., 2018).

the AugG with a hinge loss that incorporates hard negative samples. In particular, we introduce the competitive VD-oriented negative samples, which are images that contain alike visual contents related to the caption of target image, so as to enforce more distinguishable image feature predictions from the model, especially when dialog contexts are similar. In RL, we use AugG to assign reliable rewards to further improve the Questioner.

We evaluate our method on the VisDial v1.0 dataset (Das et al., 2017a). Experimental results show that our approach achieves state-of-the-art (SOTA) performance on both the image-guessing task and question diversity. Human study indicates that our Questioner generates more visually related, informative and coherent questions as compared to previous strong baselines.

Our main contributions<sup>3</sup> are concluded as follows:

- We propose a **Related entity enhanced Questioner (ReeQ)** for Visual Dialog. ReeQ generates questions using related entities as guidance and learns entity-based questioning strategy from human dialogs.
- We propose an **Augmented Guesser (AugG)** and use it to serve as an efficient component in RL to assign reliable rewards.
- We conduct experiments on the VisDial v1.0 dataset and achieve SOTA performance on both the image-guessing task and question diversity. Our Questioner outperforms previous methods on multiple criteria.

## 2 Background

GuessWhich (Das et al., 2017b) is an interactive Q-Bot-A-Bot image-guessing task. Q-Bot, who only knows the caption of an undisclosed image  $I$  at first, needs to ask a series of questions and guess the target image. A-Bot, who can see the image, answers accordingly. In this section, we formally introduce the modeling of Q-Bot and A-Bot in previous methods (Das et al., 2017b; Murahari et al., 2019a), as well as the training paradigm.

### 2.1 Model

**Q-Bot.** At round  $t$ , Q-Bot generates the question  $q_{t+1}$  and makes an image feature prediction  $\hat{y}_t$  based on the dialog history  $H_t =$

<sup>3</sup>We release our code on [https://github.com/zd11024/Entity\\_Questioner](https://github.com/zd11024/Entity_Questioner).

$\{c, (q_1, a_1), \dots, (q_t, a_t)\}$ , where  $c$  is the caption of the target image. It consists of Context Encoder, Feature Regression Network, and Question Decoder. After the dialog history  $H_t$  is encoded into a dense vector, the Feature Regression Network is used to make an image feature prediction, while the Question Decoder is used to generate question  $q_{t+1}$ .

- **Context Encoder:** The Context Encoder consists of fact encoder and history encoder, both are two-layer LSTM (Hochreiter and Schmidhuber, 1997). At round  $t$ , fact encoder encodes the question-answer pair  $(q_t, a_t)$  into the fact representation  $\mathbf{f}_t$ , then history encoder encodes  $\mathbf{f}_t$  into the history representation  $\mathbf{h}_t$ .
- **Feature Regression Network:** An MLP that uses history representation  $\mathbf{h}_t$  to make the image feature prediction  $\hat{\mathbf{y}}_t$  at round  $t$ .
- **Question Decoder:** A two-layer LSTM that decodes the question  $q_{t+1}$  given the history representation  $\mathbf{h}_t$ .

Corresponding to the two roles of Q-Bot, Context Encoder and Question Decoder form the Question Generator (QGen), while Context Encoder and Feature Regression Network form the Guesser.

**A-Bot.** Given image  $I$ , dialog history  $H_t$  and question  $q_{t+1}$ , A-Bot generates the answer  $a_{t+1}$ . A-Bot consists of a multi-modal context encoder and a decoder.

## 2.2 Training

Previous methods use a two-stage training paradigm: Q-Bot and A-Bot are firstly pre-trained through Supervised Learning (SL), then fine-tuned through Reinforcement Learning (RL).

**SL.** Q-Bot and A-Bot are respectively optimized in SL. Q-Bot (QGen and Guesser) is optimized with multi-task loss: a Cross-Entropy (CE) loss  $\mathcal{L}_{CE} = \sum_t \log(p(q_{t+1}|\mathbf{h}_t))$  to optimize QGen and a Mean Square Error (MSE) loss  $\mathcal{L}_{MSE} = \sum_t \|\mathbf{y}^{\text{gt}} - \hat{\mathbf{y}}_t\|_2^2$ , where  $\mathbf{y}^{\text{gt}}$  is the image feature of  $I$ , to optimize Guesser. A-Bot is optimized with a similar CE loss.

**RL.** Q-Bot and A-Bot are jointly optimized in RL. Q-Bot and A-Bot interact with each other and are awarded by reward  $r_t = \|\mathbf{y}^{\text{gt}} - \hat{\mathbf{y}}_t\|_2^2 - \|\mathbf{y}^{\text{gt}} - \hat{\mathbf{y}}_{t+1}\|_2^2$ . Given the Q-Bot state  $S_t^Q$  and A-Bot state  $S_t^A$ , dialog policies for Q-Bot and A-Bot

are formulated as  $\pi_{\theta_Q}(q_t|S_t^Q)$  and  $\pi_{\theta_A}(a_t|S_t^A)$ , respectively. The action of Q-Bot and A-Bot is to select next token from the vocabulary  $\mathcal{V}$ . REINFORCE(Williams, 1992) algorithm is applied to update agents' parameters with the policy gradients formulated as  $E_{\pi_Q, \pi_A} r_t \nabla_{\theta_Q} \log(\pi_Q(q_t|S_t^Q))$  and  $E_{\pi_Q, \pi_A} r_t \nabla_{\theta_A} \log(\pi_A(a_t|S_t^A))$ .

To conclude: 1) previous QGen follows a sequence-to-sequence fashion to generate questions and lacks a clear questioning strategy; and 2) reward in RL relies on a limited Guesser, that has been compromised in the eclectic training result of optimizing QGen and Guesser cooperatively.

## 3 Approach

In this section, we introduce the **Related entity enhanced Questioner (ReeQ)**, **Augmented Guesser (AugG)** and training approach. ReeQ generates questions under the guidance of related entities and learns entity-based questioning strategy from human dialogs. AugG is a strong guesser and assigns rewards during the RL optimization process.

### 3.1 Related Entity Enhanced Questioner

As illustrated in Fig. 1 (a), ReeQ consists of four modules: Context Encoder, Entity Selector, Question Decoder and Feature Regression Network. To generate a question at round  $t$ : firstly, Context Encoder encodes dialog history  $H_t$  into a history representation  $\mathbf{h}_t$ ; then, Entity Selector selects a specific entity  $e_t^*$  to ask about at this round; lastly, Question Decoder generates the question  $q_{t+1}$  with the selected entity  $e_t^*$  as guidance.

Context Encoder and Feature Regression Network are the same as previous work (see Sec. 2.1). We introduce the Entity Selector and the Question Decoder in Sec. 3.1.1 and 3.1.2, respectively.

#### 3.1.1 Entity Selector

As illustrated in Fig. 1 (a), Entity Selector contains three components, i.e., *Retriever*, *Estimator* and *Sampler*. Initially, *Retriever* retrieves a series of candidate entities using the image caption. At each round of the dialog, *Estimator* estimates a probability distribution on candidate entities w.r.t. the probable entities to ask about. Lastly, *Sampler* samples an entity based on the estimated distribution.

**Retriever.** *Retriever* uses image caption to retrieve the related entities in advance. As a prerequisite, we build entities-to-entities indexes from the entities in captions to the entities in dialogs,

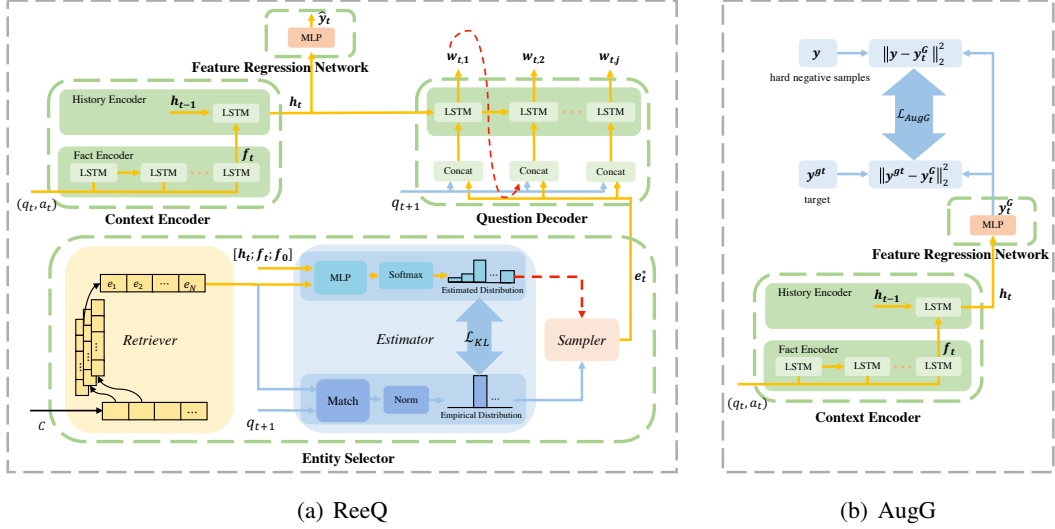


Figure 1: Illustration of our model architecture. (a) ReeQ. ReeQ contains four modules: Context Encoder, Entity Selector, Question Decoder and Feature Regression Network. Orange and blue line indicate calculation path in training. Orange and red line indicate the calculation path in inference. (b) AugG. AugG contains two modules: Context Encoder and Feature Regression Network, and is augmented with hard negative samples.

based on the co-occurrences in training data. While in use, for each dialog instance, we firstly extract the entities in caption, then use them as queries to retrieve a list of candidate entities, i.e.,  $E = \{e_1, e_2, \dots, e_N\}$ , from the established indexes. To assure the relevancy, we retain the top  $N$  entities with the highest co-occurrence frequency. More details are given in Appendix A.

**Estimator.** *Estimator* estimates a probability distribution on candidate entities, w.r.t. the probable entities to ask about at each round of the dialog.

The estimated distribution  $p_t^{est}$  is derived conditioning on current dialog, concretely the history representation  $\mathbf{h}_t$ , fact representation  $\mathbf{f}_t$  and caption representation  $\mathbf{f}_0$ . We formulate this step as:

$$v_i = \tanh([\mathbf{h}_t; \mathbf{f}_t; \mathbf{f}_0] \mathbf{W}^Q + \mathbf{e}_i \mathbf{W}^K) \mathbf{W}^A, \quad (1)$$

$$p_t^{est}(e_i) = \text{Softmax}(v_i), \quad (2)$$

where  $\mathbf{W}^Q$ ,  $\mathbf{W}^K$  and  $\mathbf{W}^A$  are learnable parameters;  $\mathbf{e}_i$  is entity representation encoded by a LSTM as an entity may include more than one word.

To learn the distribution, we establish empirical distribution  $p_t^{emp}$  from human dialog in the training data and propose an objective to encourage estimated distribution to approximate empirical distribution. In specific, empirical distribution is obtained by matching golden question  $q_{t+1}$  and candidate entities as follows:

$$p_t^{emp}(e_i) = \text{Norm}(\text{Match}(e_i, q_{t+1})), \quad (3)$$

where  $\text{Match}(e_i, q_{t+1})$  is 1 when  $e_i$  could match a sub-string of  $q_{t+1}$ , otherwise 0;  $\text{Norm}(\cdot)$  is a sum-normalization to normalize the matching result as probability distribution.

Further, we minimize the KL divergence between empirical distribution and estimated distribution throughout the dialog, so as to learn the questioning strategy from human dialog. The KL loss is formulated as:

$$\mathcal{L}_{KL} = \sum_{t,i} D_{KL}(p_t^{emp}(e_i) || p_t^{est}(e_i)). \quad (4)$$

Eq. 4 is optimized during training. While in inference, *estimator* only needs to calculate the estimated distribution.

**Sampler.** *Sampler* samples an entity based on the distribution given by *Estimator* – empirical distribution during training while estimated distribution during inference. We formulate this step as:

$$\text{sample } e_t^* \sim \begin{cases} p_t^{emp}, & \text{if training,} \\ p_t^{est}, & \text{if inference.} \end{cases} \quad (5)$$

To further refine the questioning strategy during inference, we propose a *limit-sampling rule* which limits the sampled times of each entity. In concrete, we count the sampled times  $c_t^i$  of each entity ( $c_0^i = 0$ ), and when  $c_t^i$  reaches the upper bound  $B$ , the corresponding entity will be masked. Accordingly, the refined estimated distribution is:

$$p_t^{est}(e_i) = \text{MaskedSoftmax}(I[c_t^i < B]v_i). \quad (6)$$

The sampled times is updated as follows:

$$\hat{c}_{t+1}^i = \hat{c}_t^i + I[e_t^* = e_i]. \quad (7)$$

where  $I[\cdot]$  equals 1 when the expression in square brackets is true, else 0.

### 3.1.2 Question Decoder

Question Decoder is a two-layer LSTM that generates next question using the selected entity as a hint. At each time step  $j$ , we concatenate the previously generated word embedding  $\mathbf{w}_{t,j-1}$  with the selected entity representation  $\mathbf{e}_t^*$  as input.

With  $\mathbf{h}_{t,j}^D$  ( $\mathbf{h}_{t,0}^D = \mathbf{h}_t$ ) denoting the hidden states of the decoder at the time step  $j$ , we formulate the decoding step as:

$$\mathbf{h}_{t,j}^D = LSTM^D([\mathbf{w}_{t,j-1}; \mathbf{e}_t^*], \mathbf{h}_{t,j-1}^D), \quad (8)$$

$$p(w_{t,j} | \mathbf{h}_{t,j}^D) = softmax(\mathbf{h}_{t,j}^D \mathbf{W}^D). \quad (9)$$

## 3.2 Augmented Guesser

We establish the Augmented Guesser (AugG) using the same two modules, i.e., Context Encoder and Feature Regression Network, as shown in Fig. 1 (b). At round  $t$ , given dialog history  $H_t$ , AugG makes the image feature prediction  $\mathbf{y}_t^G$ .

To enable a strong AugG, we provide two types of negative samples during training. The first is the VD-oriented negative samples, which are images retrieved by the distinctive caption in each dialog instance (see details in Appendix B). Thus, the VD-oriented negative image samples have alike visual semantics with the target image. Such negative samples enforce more distinguishable image predictions under similar dialog context. The second is the stochastic negative samples in mini-batch, drawing on the use of negative mining in other tasks (Schroff et al., 2015; Manmatha et al., 2017; Faghri et al., 2018). Sec. 3.3.1 introduces the detailed loss function.

## 3.3 Training

Our training is two-stage: 1) firstly train ReeQ, AugG and A-Bot through Supervised learning (SL); then 2) jointly fine-tune ReeQ and A-Bot through Reinforcement Learning (RL) with the reward assigned by AugG.

### 3.3.1 Supervised Learning

**Training for ReeQ.** Similar to previous work, ReeQ is optimized with multi-task loss that includes  $\mathcal{L}_{CE}$  and  $\mathcal{L}_{MSE}$  as in Sec. 2.2. Besides, as given in Eq. 4,  $\mathcal{L}_{KL}$  is to make the estimated

distribution approximate the empirical distribution. Thus, the loss function for ReeQ is:

$$\mathcal{L}_{ReeQ} = \mathcal{L}_{CE} + \beta \mathcal{L}_{MSE} + \gamma \mathcal{L}_{KL}, \quad (10)$$

where  $\beta$  and  $\gamma$  are hyper-parameters.

**Training for AugG.** The loss to optimize AugG is based on  $\alpha$ -margin max-of-hinges loss (Faghri et al., 2018) and incorporates two types of negative samples (Sec. 3.2). We formulate the loss as:

$$\begin{aligned} \mathcal{L}_{AugG} = & \sum_t \max_{y \in Y} [\alpha + \|\mathbf{y}^{gt} - \mathbf{y}_t^G\|_2^2 - \|\mathbf{y} - \mathbf{y}_t^G\|_2^2]_+ \\ & + \sum_t \max_{y' \in Y'} [\alpha + \|\mathbf{y}^{gt} - \mathbf{y}_t^G\|_2^2 - \|\mathbf{y}' - \mathbf{y}_t^G\|_2^2]_+, \end{aligned} \quad (11)$$

where  $[\cdot]_+ = \max(0, \cdot)$ , set  $Y$  consists of the VD-oriented negative samples and  $Y'$  consists of the stochastic negative samples.

**Training for A-Bot.** The training of A-Bot is the same as in Sec. 2.2.

### 3.3.2 Reinforcement Learning

In RL, Q-Bot and A-Bot are jointly optimized with the reward assigned by AugG. At round  $t$ , AugG makes the image feature prediction  $\mathbf{y}_t^G$ . Then as Q-Bot questions  $q_{t+1}$  and A-Bot answers  $a_{t+1}$ , AugG predicts  $\mathbf{y}_{t+1}^G$ . Accordingly, Q-Bot and A-Bot are awarded by the reward:

$$r_t^G = \|\mathbf{y}^{gt} - \mathbf{y}_t^G\|_2^2 - \|\mathbf{y}^{gt} - \mathbf{y}_{t+1}^G\|_2^2. \quad (12)$$

## 4 Experiments

We evaluate our method on the large-scale VisDial v1.0 dataset (Das et al., 2017a), where the train split contains 123,287 images and the validation split contains 2,064 images, and each image has the corresponding caption and 10-round dialog.

For training details, please refer to Appendix C. The upper bound  $B$  in the *sampler* of ReeQ’s Entity Selector is set to 1, and we discuss its effect and more options in Appendix D.

### 4.1 Comparing Methods

We compare our method with previous strong baselines. To clarify, we introduce the comparing methods in QGen and Guesser, w.r.t. the roles to generate questions or make image predictions.

**QGen:** 1) DasQ (Das et al., 2017b): the baseline method; 2) DivQ (Murahari et al., 2019a): an improved method that penalizes the similarity of successive encoded dialog hidden states to encourage question diversity; 3) ReeQ: our method.

#		QGen	Guesser	MRR $\uparrow$	R@1 $\uparrow$	R@5 $\uparrow$	R@10 $\uparrow$	Mean $\downarrow$	PMR $\uparrow$
1	SL	DasQ $\dagger$	DasG $\dagger$	7.80	2.56	9.49	17.87	127.84	93.83
2		DivQ $\dagger$	DivG $\dagger$	10.73	3.39	14.82	25.29	87.92	95.73
3		DasQ $\dagger$	AugG	25.65	16.30	40.50	55.43	28.57	98.76
4		DivQ $\dagger$	AugG	31.59	19.14	44.96	59.06	22.03	98.93
5		ReeQ	AugG	31.21	17.78	45.01	59.98	20.60	99.00
6	RL	DasQ $\dagger$	DasG $\dagger$	7.54	2.18	9.78	17.05	125.07	93.94
7		DivQ $\dagger$	DivG $\dagger$	10.79	3.39	15.69	25.33	89.28	95.67
8		DasQ $\dagger$	AugG	29.52	16.57	42.68	57.99	25.36	98.77
9		DivQ $\dagger$	AugG	31.08	17.93	44.91	60.41	22.35	98.91
10		ReeQ	AugG	<b>33.65</b>	<b>19.91</b>	<b>48.50</b>	<b>62.94</b>	<b>18.05</b>	<b>99.13</b>

Table 2: Comparing results on image-guessing task.  $\dagger$  represents that the evaluated models are from (Murahari et al., 2019b).  $\uparrow$  indicates higher is better.  $\downarrow$  indicates lower is better.

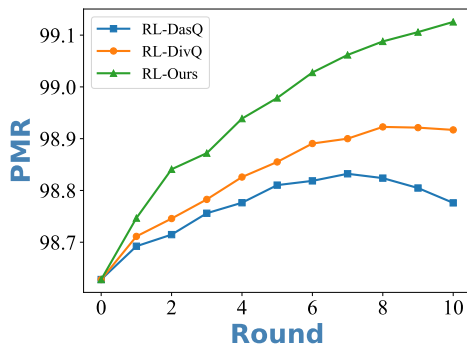


Figure 2: Trends of PMR as dialog progresses.

**Guesser:** 1) DasG: Guesser that is cooperatively trained with DasQ; 2) DivG: Guesser that is cooperatively trained with DivQ; 3) AugG: our Augmented Guesser.

## 4.2 Quantitative Results

**Image-Guessing Task.** We evaluate the performance on image-guessing task. In concrete, QGen and A-Bot firstly generate 10-round dialog, then Guesser makes a prediction about the unseen image, lastly the candidate images (images in validation split) are sorted according to their similarity to the prediction and compute the rank of the target image. The evaluation metrics are: 1) MRR (Radev et al., 2002): mean reciprocal rank of target image; 2) R@k (Das et al., 2017b): the existence of target image in the top-k images; 3) Mean (Das et al., 2017b): mean rank of target image; 4) PMR (Das et al., 2017b): percentile mean rank.

We illustrate the results in Tab. 2. As shown in row 10, our method achieves the best performance on all metrics and becomes the new state of the art, with a MRR of 33.65, R@1 of 19.91, R@5 of 48.50, R@10 of 62.94 and PMR of 99.13. To make

fair comparisons, we further use the same AugG as Guesser to evaluate all methods, as shown in row 8, 9 and 10. As can be seen, our method is superior than RL-DasQ and RL-DivQ on all metrics. As in SL (see row 3,4 and 5), SL-ReeQ outperforms other methods on R@5, R@10, Mean and PMR, but does not surpass SL-DivQ on MRR and R@1. This may come from the accumulated errors of entity selection in the instances that are unseen in SL, as the training data only covers limited selecting trajectories. And since RL enables more explorations, the problem is relieved and RL-ReeQ achieves the best performance.

Fig. 2 shows the trends of PMR in the 10-round dialog. To make a fair comparison, we use the same AugG to serve as the Guesser. As can be seen, only our method enables the continuously increasing image-guessing performance as dialog progresses. The trends indicate that our method can generate the constantly visually-related and informative dialogs while others cannot.

**Question Diversity.** We evaluate the question diversity of Q-Bot with the following metrics: 1) Unique questions (Murahari et al., 2019a): mean number of unique questions in the 10-round dialog; 2) Mutual overlap (Deshpande et al., 2018): mean BLEU-4 (Papineni, 2002) overlap with the other 9 questions in the 10-round dialog; 3) Dist-n and Ent-n (Li et al., 2016; Zhang et al., 2018): number and entropy of distinct n-grams in the generated questions divided by the total number of tokens.

As shown in Tab. 3, row 6 indicates that our method achieves the new SOTA performance on question diversity. Specifically, our RL-ReeQ achieves approximately 2 points improvement on Unique questions, which shows that we have greatly reduced repetition (row 4, 5 vs. row 6).

#			Unique questions $\uparrow$	Mutual overlap $\downarrow$	Dist-1 $\uparrow$	Dist-2 $\uparrow$	Ent-1 $\uparrow$	Ent-2 $\uparrow$
1		DasQ $\ddagger$	6.57	0.60	2.70	3.00	0.34	0.42
2	SL	DivQ $\ddagger$	7.45	0.51	2.82	3.18	0.38	0.48
3		ReeQ	<b>9.97</b>	<b>0.11</b>	2.87	3.41	<b>0.46</b>	0.63
4		DasQ $\ddagger$	6.70	0.58	2.72	3.03	0.35	0.43
5	RL	DivQ $\ddagger$	8.19	0.41	2.90	3.31	0.40	0.53
6		ReeQ	<b>9.97</b>	<b>0.11</b>	<b>2.90</b>	<b>3.45</b>	<b>0.46</b>	<b>0.64</b>

Table 3: Question diversity on VisDial v1.0 val.  $\ddagger$  means the results are cited from (Murahari et al., 2019a).  $\uparrow$  indicates higher is better.  $\downarrow$  indicates lower is better.

	NDCG $\uparrow$	MRR $\uparrow$	R@1 $\uparrow$	R@5 $\uparrow$	R@10 $\uparrow$	Mean $\downarrow$
SL $\ddagger$	53.10	46.21	36.11	55.82	62.22	19.58
RL $\ddagger$	53.76	46.35	36.22	56.15	62.41	<b>19.34</b>
RL-Div $\ddagger$	53.91	46.46	36.31	56.26	62.53	19.35
RL-ReeQ	<b>54.35</b>	<b>46.52</b>	<b>36.45</b>	<b>56.34</b>	<b>62.68</b>	19.56

Table 4: A-Bot performance on VisDial v1.0 val.  $\ddagger$  means the results are cited from (Murahari et al., 2019a).  $\uparrow$ : higher is better.  $\downarrow$ : lower is better.

	MRR $\uparrow$	R@1 $\uparrow$	R@5 $\uparrow$	R@10 $\uparrow$	Mean $\downarrow$	PMR $\uparrow$
DasG $\ddagger$	3.29	11.58	20.06	9.17	108.11	94.76
AugG $^-$	32.52	18.94	47.38	62.79	18.47	99.10
AugG	<b>33.63</b>	<b>20.06</b>	<b>47.97</b>	<b>63.23</b>	<b>17.72</b>	<b>99.14</b>

Table 5: Guesser performance on VisDial v1.0 val.  $\ddagger$  represents that the evaluated model are from (Murahari et al., 2019b). AugG $^-$  means only stochastic negative samples are used in training.

Noticeably, our result on Unique questions is approaching the upper bound, i.e., 10. Besides, our method also achieves better language diversity according to Mutual overlap, Dist-1, Dist-2, Ent-1 and Ent-2 (row 3 and row 6).

**A-Bot Performance.** We evaluate the A-Bot performance in a retrieval setting, following Das et al. (2017a). Additional 100 candidate answers for each instance are provided and the model is evaluated by retrieval metrics: 1) NDCG (Järvelin and Kekäläinen, 2002): normalized discounted cumulative gain; 2) MRR (Radev et al., 2002): mean reciprocal rank of the ground truth answer; 3) R@k (Das et al., 2017a): the existence of the ground truth answer in the top-k answers; 4) Mean (Das et al., 2017a): mean rank of the ground truth answer. Tab. 4 shows the comparing results of A-Bot performance. Our model achieves higher NDCG, MRR, R@1, R@5 and R@10.

**Guesser Performance.** Guesser performance is tested on the given ground-truth dialog, shown in

#		MRR $\uparrow$	R@1 $\uparrow$	R@5 $\uparrow$	R@10 $\uparrow$	Mean $\downarrow$	PMR $\uparrow$
1	DasQ+r $_1$	25.65	16.30	40.50	55.43	28.57	98.76
2	DasQ+r $_2$	32.19	19.09	46.27	60.76	21.64	98.95
3	DasQ+r $_3$	32.77	19.47	46.75	62.89	20.45	99.01
4	ReeQ+r $_1$	32.27	18.56	46.75	61.01	19.58	99.05
5	ReeQ+r $_2$	32.78	19.38	47.00	62.65	19.46	99.06
6	ReeQ+r $_3$	<b>33.65</b>	<b>19.91</b>	<b>48.5</b>	<b>62.94</b>	<b>18.05</b>	<b>99.13</b>

Table 6: Performance of ablation methods on image-guessing task.  $r_1$ ,  $r_2$  and  $r_3$  represent the reward is assigned by the cooperatively optimized guesser, AugG $^-$  and AugG, respectively.

Tab. 5. As can be seen, AugG achieves the best performance with a PMR of 99.14. By comparing AugG with AugG $^-$ , we see that the performance is improved by VD-oriented negative samples.

**Ablation Study.** We conduct ablation study to investigate the effect of ReeQ and the effect of rewards given by different Guessers, respectively. We use AugG as Guesser and evaluate the further image-guessing performance for fair comparisons. As shown in Tab. 6, we have following observations: 1) by comparing the results in upper part and lower part, we see the superiority of ReeQ; 2) in each part, by comparing the results among + $r_1$ , + $r_2$  and + $r_3$ , we see the respective improvements brought by separately optimized Guesser, as well as VD-oriented hard negatives in training Guesser. This indicates that our Guesser assigns more reliable rewards that help achieve improved performance in image-guessing.

### 4.3 Qualitative Results

Fig. 3 presents an example of generated dialogs from comparing methods. As shown, both RL-DasQ and RL-DivQ ask repetitive questions while ours asks non-repetitive questions. Moreover, the questions generated by our method are more informative, detailed and of higher relevance to target image. Noticeably, we find our method generates

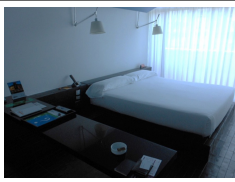
Image + Caption	RL-DasQ	RL-DivQ	RL-Ours
 <p>Caption: an empty room with a desk and a bed</p>	<p>is the photo in color ? yes  is the photo close up ? no  do you see anyone ? no  is the bed made ? yes  what color is the bedding ? white  is the bed made ? yes  what color is the bedding ? white  do you see any pillows ? yes  do you see any windows ? no</p>	<p>is the photo in color ? yes  do you see any people ? no  is the bed made ? yes  what color is the bedding ? white  do you see a window ? no  do you see a dresser ? no  do you see a window ? no  is the room well lit ? yes  do you see a window ? no  is the room well lit ? yes</p>	<p>is the photo in color? yes  is there any <b>people</b> ? no  is the <b>bed</b> made ? yes  what color is the <b>bedspread</b> ? white  is there <b>pillows</b> on the bed ? yes  what color are the <b>walls</b> ? white  is there a <b>window</b> ? yes  is there <b>curtains</b> on the window ? no  is there a <b>desk</b> ? no  can you see the <b>ceiling</b> ? yes</p>

Figure 3: An example of dialogs on VisDial v1.0 val. Red marks the entity selected at current round.

	R@1 $\uparrow$	R@5 $\uparrow$
Human	75.00	95.00
RL-DasQ	27.00	80.33
RL-DivQ	42.67	83.33
RL-Ours	46.33	89.00

Table 7: Results on image-guessing in human study.

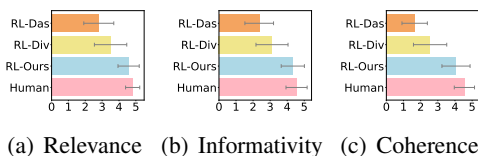


Figure 4: Results on dialog quality in human study, including means and variances.

questions that are coherent with the selected entities (marked in red), indicating the entities guide the question generation effectively. We also see the sequential entities follow a clear strategy. For example, it asks “bed” at round 3, then “bedspread” and “pillows on the bed”. Afterwards, it asks other furnishings in the room successively. More qualitative results are given in Appendix E.

#### 4.4 Human Study

We conduct human studies to further evaluate the dialog generated by different methods, i.e. Human, RL-DasQ, RL-DivQ and RL-Ours. Six postgraduate students are recruited and each one evaluates 50 instances for each method.

**Image-guessing Task.** Das et al. (2017b) design the task to evaluate how human-understandable and image-discriminative the generated dialogs are. Evaluators are required to pick the top-1 and top-5 likely images from a 16-candidate-image pool, including 1 target, 5 nearest neighbors and 10 random ones. As in Tab. 7, our method outperforms previous work while has a gap with Human.

**Dialog Quality.** Evaluators score the dialogs based on the dialog history and image. The scoring adopts a 5-point scale, which is evaluated in terms of relevance, informativity and coherence. Relevance indicates how well the generated dialogs are related to the target image and the caption. Informativity measures whether the dialog provides sufficient information related to the target image. Coherence assesses whether the generated dialogs are less repetitive, natural and coherent. As in Fig. 4, humans judge our method as generating more visually related, informative and coherent dialogs than other methods.

## 5 Discussion

**Entity-Selection Accuracy.** Considering the diverse questioning strategies in real-world scenes, “correct” entity is hard to define. Therefore, we conduct human study. We sample 50 generated dialogs (selected entities highlighted) in val-set and ask 3 evaluators to judge whether entities are relevant to the image and caption (i.e., are correct in the current context). 92.5% selected entities are regarded as relevant (qualitatives in Appendix E). Additionally, we study the ability to select entities just the same as entities in ground-truth human dialogs. Evaluation result on val-set shows 14% selected entities are same, indicating the model has learned from human dialogs since there are 100 candidates. 14% is not high, but it is reasonable considering the rich visual scenes and various questioning paths.

**Computational Cost.** Between ReeQ and DasQ, the ratio of time to get the best performed model is 1.47 (11h vs. 7.5h) in SL and 1.38 (14.5h vs. 10.5h) in RL. At inference, ReeQ spends 1.1 times the baseline (72s vs. 64s). We conclude ReeQ costs more for estimating the entities to ask and generating entity-guided questions. Despite additional time cost, generation results of ReeQ are inspiring.



## 6 Related Work

Our work is mostly related to building open-domain Visual Dialog Questioner in the image-guessing task setting. Das et al. (2017b) propose the task and generate questions in a sequence-to-sequence fashion. Murahari et al. (2019a) propose to reduce repetition by penalizing the similarity in successive dialog hidden states. Zhou et al. (2019) retrieve the most-likely image, encode the image into a multi-modal context vector and use it to decode questions. These methods follow a sequence-to-sequence fashion while ReeQ explicitly uses related-entities as guidance to generate questions following a learned strategy.

Our work is also relevant to the works (Zhang et al., 2017; Zhao and Tresp, 2018; Strub et al., 2017; Shekhar et al., 2019; Shukla et al., 2019; Xu et al., 2020) that focus on VD Questioner for Guess-What?! (de Vries et al., 2017), where the goal is to locate a target object in the image and the answers can only be “yes/no/not available”. Compared to them, building a Questioner in a more open-domain VD setting is of more difficulty. Moreover, Q-Bot in GuessWhich has no access to visual information, making it harder to ask visually related questions.

## 7 Conclusion

In this paper, we propose **Related entity enhanced Questioner (ReeQ)** and **Augmented Guesser (AugG)** to enhance Visual Dialog Questioner in both SL and RL. ReeQ generates questions with related entities as guidance and learns an entity-based questioning strategy from human dialogs. AugG is a strong Guesser that is optimized for VD especially. We use AugG to assign reliable rewards in RL. Experimental results on VisDial v1.0 show our method outperforms priors on multiple criteria.

## Acknowledgements

We would like to thank anonymous reviewers for their suggestions and comments. The work was supported by the National Natural Science Foundation of China (NSFC62076032) and the Cooperation Project with Beijing SanKuai Technology Co., Ltd.

## References

Shubham Agarwal, Trung Bui, Joon-Young Lee, Ioannis Konstas, and Verena Rieser. 2020. [History for visual dialog: Do we really need it?](#) In *Proceedings*

*of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8182–8197, Online. Association for Computational Linguistics.

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 *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 6077–6086. IEEE Computer Society.

Feilong Chen, Xiuyi Chen, Fandong Meng, Peng Li, and Jie Zhou. 2021. [GoG: Relation-aware graph-over-graph network for visual dialog](#). In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 230–243, Online. Association for Computational Linguistics.

Feilong Chen, Fandong Meng, Jiaming Xu, Peng Li, Bo Xu, and Jie Zhou. 2020. [DMRM: A dual-channel multi-hop reasoning model for visual dialog](#). In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 7504–7511. AAAI Press.

Abhishek Das, Satwik Kottur, Khushi Gupta, Avi Singh, Deshraj Yadav, José M. F. Moura, Devi Parikh, and Dhruv Batra. 2017a. [Visual dialog](#). In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 1080–1089. IEEE Computer Society.

Abhishek Das, Satwik Kottur, José M. F. Moura, Stefan Lee, and Dhruv Batra. 2017b. [Learning cooperative visual dialog agents with deep reinforcement learning](#). In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, pages 2970–2979. IEEE Computer Society.

Harm de Vries, Florian Strub, Sarath Chandar, Olivier Pietquin, Hugo Larochelle, and Aaron C. Courville. 2017. [Guesswhat?! visual object discovery through multi-modal dialogue](#). In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 4466–4475. IEEE Computer Society.

Aditya Deshpande, Jyoti Aneja, Liwei Wang, Alexander G. Schwing, and David A. Forsyth. 2018. [Diverse and controllable image captioning with part-of-speech guidance](#). *CoRR*, abs/1805.12589.

Fartash Faghri, David J. Fleet, Jamie Ryan Kiros, and Sanja Fidler. 2018. [VSE++: improving visual-semantic embeddings with hard negatives](#). In *British Machine Vision Conference 2018, BMVC 2018, Newcastle, UK, September 3-6, 2018*, page 12. BMVA Press.

- Zhe Gan, Yu Cheng, Ahmed 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, Florence, Italy. Association for Computational Linguistics.
- S. Hochreiter and J. Schmidhuber. 1997. Long short-term memory. *Neural Computation*, 9(8):1735–1780.
- Kalervo Järvelin and Jaana Kekäläinen. 2002. [Cumulated gain-based evaluation of ir techniques](#). *ACM Trans. Inf. Syst.*, 20(4):422–446.
- R. Krishna, Y. Zhu, O. Groth, J. Johnson, and F. F. Li. 2017. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision*, 123(1).
- Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. 2016. [A diversity-promoting objective function for neural conversation models](#). In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 110–119, San Diego, California. Association for Computational Linguistics.
- X. Li, X. Yin, C. Li, P. Zhang, X. Hu, L. Zhang, L. Wang, H. Hu, L. Dong, and F. Wei. 2020. *Oscar: Object-Semantics Aligned Pre-training for Vision-Language Tasks*.
- Rongzhong Lian, Min Xie, Fan Wang, Jinhua Peng, and Hua Wu. 2019. [Learning to select knowledge for response generation in dialog systems](#). In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019*, pages 5081–5087. ijcai.org.
- 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 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 314–324.
- R. Manmatha, Chao-Yuan Wu, Alexander J. Smola, and Philipp Krähenbühl. 2017. [Sampling matters in deep embedding learning](#). In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, pages 2859–2867. IEEE Computer Society.
- Nirbhay Modhe, Viraj Prabhu, Michael Cogswell, Satwik Kottur, Abhishek Das, Stefan Lee, Devi Parikh, and Dhruv Batra. 2018. [Visdial-rl-pytorch](https://github.com/batra-mlp-lab/visdial-rl.git). <https://github.com/batra-mlp-lab/visdial-rl.git>.
- Vishvak Murahari, Prithvijit Chattopadhyay, Dhruv Batra, Devi Parikh, and Abhishek Das. 2019a. [Improving generative visual dialog by answering diverse questions](#). In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1449–1454, Hong Kong, China. Association for Computational Linguistics.
- Vishvak Murahari, Prithvijit Chattopadhyay, Dhruv Batra, Devi Parikh, and Abhishek Das. 2019b. [Visdial-div-pytorch](https://github.com/vmurahari3/visdial-diversity.git). <https://github.com/vmurahari3/visdial-diversity.git>.
- V. Q. Nguyen, M. Suganuma, and T. Okatani. 2020. Efficient attention mechanism for visual dialog that can handle all the interactions between multiple inputs. *Springer, Cham*.
- Yulei Niu, Hanwang Zhang, Manli Zhang, Jianhong Zhang, Zhiwu Lu, and Ji-Rong Wen. 2019. [Recurisive visual attention in visual dialog](#). In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 6679–6688. Computer Vision Foundation / IEEE.
- K. Papineni. 2002. A method for automatic evaluation of machine translation. *Proc Acl*.
- Dragomir R. Radev, Hong Qi, Harris Wu, and Weiguo Fan. 2002. [Evaluating web-based question answering systems](#). In *Proceedings of the Third International Conference on Language Resources and Evaluation (LREC’02)*, Las Palmas, Canary Islands - Spain. European Language Resources Association (ELRA).
- Florian Schroff, Dmitry Kalenichenko, and James Philbin. 2015. [Facenet: A unified embedding for face recognition and clustering](#). In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*, pages 815–823. IEEE Computer Society.
- Ravi Shekhar, Aashish Venkatesh, Tim Baumgärtner, Elia Bruni, Barbara Plank, Raffaella Bernardi, and Raquel Fernández. 2019. [Beyond task success: A closer look at jointly learning to see, ask, and Guess-What](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2578–2587, Minneapolis, Minnesota. Association for Computational Linguistics.
- Pushkar Shukla, Carlos Elmadjian, Richika Sharan, Vivek Kulkarni, Matthew Turk, and William Yang Wang. 2019. [What should I ask? using conversationally informative rewards for goal-oriented visual dialog](#). In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6442–6451, Florence, Italy. Association for Computational Linguistics.

- Florian Strub, Harm de Vries, Jérémie Mary, Bilal Piot, Aaron C. Courville, and Olivier Pietquin. 2017. [End-to-end optimization of goal-driven and visually grounded dialogue systems](#). In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI 2017, Melbourne, Australia, August 19-25, 2017*, pages 2765–2771. [ijcai.org](http://ijcai.org).
- R. J. Williams. 1992. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, 8(3-4):229–256.
- Z. Xu, F. Feng, X. Wang, Y. Yang, and H. Jiang. 2020. Answer-driven visual state estimator for goal-oriented visual dialogue. In *MM '20: The 28th ACM International Conference on Multimedia*.
- Junjie Zhang, Qi Wu, Chunhua Shen, Jian Zhang, Jianfeng Lu, and Anton van den Hengel. 2017. [Asking the difficult questions: Goal-oriented visual question generation via intermediate rewards](#). *CoRR*, abs/1711.07614.
- Yizhe Zhang, Michel Galley, Jianfeng Gao, Zhe Gan, Xiujun Li, Chris Brockett, and Bill Dolan. 2018. [Generating informative and diverse conversational responses via adversarial information maximization](#). In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pages 1815–1825.
- Rui Zhao and Volker Tresp. 2018. [Learning goal-oriented visual dialog via tempered policy gradient](#). In *2018 IEEE Spoken Language Technology Workshop (SLT)*, pages 868–875.
- Mingyang Zhou, Josh Arnold, and Zhou Yu. 2019. [Building task-oriented visual dialog systems through alternative optimization between dialog policy and language generation](#). In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 143–153, Hong Kong, China. Association for Computational Linguistics.

## A Related Entity Retrieval

This part we introduce related entity retrieval referred in Sec. 3.1.1. As a prerequisite, We use the object dictionary in Visual Genome (Krishna et al., 2017) as our entity vocabulary, and build entities-to-entities indexes from the entities in captions to the entities in dialogs. Then follows 4 steps:

- 1) Extract caption entities  $E_c$  in the caption  $c$  through template matching.
- 2) Retrieve probable entities  $E_p$  by using entities in  $E_c$  as queries from the established indexes.
- 3) Sort entities in  $E_p$  according to the sum of co-occurrence frequency with the entities in  $E_c$ .
- 4) Retain the top- $N$  entities to form a candidate entity set  $E = \{e_1, e_2, \dots, e_N\}$ .

As for details, We empirically set  $N$  to 100. Averagely, 6.4 questions per 10-round dialog in the train split could match the candidate entities while 6.1 in the validation split. *Selector* will choose an additional ‘NULL’ when no entity could match with the question.

## B VD-oriented Negative Samples

We obtain VD-oriented negative samples through the following steps. Firstly, we build objects-to-images indexes through objects in the image, which are extracted using bottom-up-attention (Anderson et al., 2018). Secondly, we retrieve top-100 images through the index and the pre-trained model (we use OSCAR (Li et al., 2020)) successively. Lastly, we sample 8 images from the retrieved images to form the VD-oriented negative samples, learning from prior work (Lian et al., 2019).

## C Training Details

We implement our method with Pytorch and conduct all experiments on NVIDIA Tesla V100 GPU.

Overall, we follow the same training methods with previous work. In SL, we pre-train ReeQ for 15 epochs. We use Adam optimizer with a mini-batch size of 20 and a learning rate of 1e-3 decayed to 5e-5.  $\beta$  and  $\gamma$  are set to be 1000 and 1. We also apply Dropout rate of 0.5 before the feature regression network as previous work. For AugG, we train AugG for 10 epochs and select the best performed model on the validation set. Adam optimizer is used with a learning rate of 1e-3 and a batch size of 20. The margin  $\alpha$  in  $\mathcal{L}_{AugG}$  is set to 0.1 empirically. And we directly use the released

	Image-guessing		Question diversity	
	PMR	MRR	Unique questions	Mutual overlap
$w/B = 1$	99.13	33.65	9.97	0.11
$w/B = 2$	99.06	33.50	9.18	0.25
$w/B = 3$	99.06	33.31	9.03	0.26
$w/B = \infty$	99.05	33.03	8.87	0.28

Table 8: The effect of limit-sampling rule on image-guessing task.  $w/B = n$  means that the upper bound is set to be  $n$  when ReeQ generates questions in inference.

checkpoint of A-Bot from (Murahari et al., 2019b). In RL, we apply the same curriculum learning to fine-tune the model. Specifically, we use SL in the first  $K$  rounds of dialog, and optimize the model through RL in the remaining 10- $K$  rounds. We start with  $K=9$  and gradually decrease to  $K=4$ , and fine-tune the model for 12 epochs with a mini-batch size of 32. After each epoch, the model with the maximum PMR is selected for evaluation.

## D Effect of Limit-sampling Rule

We investigate the effect of the limit-sampling rule. In Tab. 8, when the upper bound  $B$  (the maximum sampled times of each entity, as in Eq. 7) increases, the performance of image-guessing task and question diversity becomes lightly worse, validating the efficiency of limit-sampling rule in inference. Besides, we study the absolute ability that ReeQ has learned through training with  $w/B = \infty$ , which is equivalent to no limit-sampling rule is used in inference. As can be seen, ReeQ still achieves high unique questions of 8.87 and PMR of 99.05, indicating our ReeQ has acquired the ability to ask non-repetitive and visually related questions following the entity-based questioning strategy during training. And the limit-sampling rule further avoids the repetition through controlling the maximum sampled times of each entity, benefited from ReeQ following an entity-to-question fashion to generate questions.

## E Qualitative Examples

Fig. 5 gives more examples of dialogs generated by different methods. RL-DasQ usually asks repetitive questions. RL-DivQ reduces repetition while generates less visually related questions. Comparatively, ours asks more informative and visually related questions. As marked in red, our method can select appropriate entity at each round and ask question accordingly.

Image+ Caption	RL-DasQ	RL-DivQ	RL-Ours
 <p>Caption: 2 people sitting in the snow with a snowboard on</p>	<p>is the photo in color ? yes  is the photo close up ? no  what gender are the people ? they are far away , i can't tell  what are they wearing ? ski gear  do you see a ski lift ? no  do you see trees ? yes  do you see other people ? no  do you see a ski lift ? no  do you see trees ? no  do you see a ski lift ? no</p>	<p>is the photo in color ? yes  is the photo close up ? no  what gender are the people ? can't tell  are they wearing goggles ? can't tell  are they wearing hats ? yes  are they wearing goggles ? can't tell  what color is the hat ? black  do you see other people ? no  do you see a ski lift ? no  do you see trees ? yes</p>	<p>what gender are the <b>people</b> ? they are far away , i can't tell  what are they wearing ? snow pants and ski jackets  what color is the <b>ski lift</b> ? black and white  what color is the <b>snow</b> ? white  is it on a <b>mountain</b> ? yes  any <b>trees</b> ? yes  any <b>clouds</b> ? no  is the <b>hill</b> steep ? yes  what color are the <b>gloves</b> ? black  what color is the <b>snowboard</b> ? white</p>
 <p>Caption: a bathroom with wooden sink and plastic shower curtain</p>	<p>is the photo in color ? yes  is there any people ? no  is this indoors or outdoors ? indoors  is there a window ? yes  is there a shower curtain ? no  is there a shower curtain ? no  is there a shower curtain ? no  is there a shower curtain ? no  is there a shower curtain ? no  is there a shower curtain ? no</p>	<p>is the photo in color ? yes  is there any people ? no  is this at a business or a person 's house ? house  is there a mirror ? no  is there a shower curtain ? no  is there a shower curtain ? no  is there a window ? no  is there a toilet ? no  is there a window ? no  is there a toilet ? no</p>	<p>what color are the <b>fixtures</b> ? silver  is the photo well lit ? yes  can you see a <b>mirror</b> ? no  can you see a <b>tub</b> ? no  can you see a <b>toilet</b> ? yes  what color are the <b>walls</b> ? white  is there a <b>rug</b> ? no  are there any <b>towels</b> ? no  what color is the <b>sink</b> ? white  what color is the <b>cabinet</b> ? brown</p>
 <p>Caption: the red stripe bullet train waits in the train station for passengers</p>	<p>is the photo in color ? yes  is the photo close up ? no  do you see the conductor ? no  do you see the conductor ? no  do you see the conductor ? no  do you see the conductor ? no  do you see the conductor ? no  do you see the conductor ? no  do you see the conductor ? no  do you see the conductor ? no</p>	<p>is this in color ? yes  do you see any people ? no  is this indoors ? yes  what color is the chair ? black  do you see any other animals ? no  what color is the chair ? black  do you see the floor ? yes  what color is the floor ? i can't see the floor  what color are the walls ? white  do you see the floor ? no</p>	<p>is the train at a <b>station</b> ? yes  can you see the <b>conductor</b> ? no , i can't see the driver  is there <b>writing</b> on the train ? yes , but i can't read  are there <b>lines</b> on the train ? yes  can you see any <b>signs</b> ? no signs  can you see any <b>people</b> ? yes , there are people in the background  can you see the <b>sky</b> ? yes  what is the weather like ? is it sunny  can you see the <b>tracks</b> ? yes  can you see any <b>gravel</b> ? no , i can't see the ground</p>
 <p>Caption: the man catches a small wave on his surfboard</p>	<p>is the photo in color ? no  how old is the man ? in his 20 's  what race is the man ? white  what color is his hair ? black  is he wearing glasses ? no  what color is his hair ? black  what color is his surfboard ? white  what color is his surfboard ? white  can you see the sky ? yes  is it sunny ? yes</p>	<p>how old is the man ? i can't see his face , i don't know  what color is his hair ? i can't tell , he 's wearing a hat  what color is the wetsuit ? black  what color is the surfboard ? i can't tell  is it sunny out ? yes  can you see the sky ? yes  are there any clouds ? no  are there any other people ? no  what color is the man 's hair ? i can't tell  is the man wearing a wetsuit ? yes</p>	<p>how old is the <b>man</b> ? he looks to be in his 20s  are there other people ? no  what is he wearing ? a wetsuit  what color is the <b>surfboard</b> ? i can't tell it 's a black and white photo  is he wearing a <b>wetsuit</b> ? yes  is the <b>water</b> calm ? yes  can you see the <b>shore</b> ? no  are there any <b>boats</b> ? no  is the <b>sun</b> shining ? i can't tell , it 's a black and white photo  are there any other <b>people</b> surfing ? no</p>
 <p>Caption: a bundled up family is walking on a snow covered sidewalk lined with trees</p>	<p>is the photo in color ? yes  is the photo close up ? no  how many people ? 2  what are they wearing ? shirts and pants  do you see a ski lift ? yes  do you see any trees ? no  do you see any buildings ? yes  do you see any trees ? no  do you see any buildings ? yes  do you see any trees ? no</p>	<p>is it snowing ? no  how many people are there ? 3  is it sunny ? no  how many children are there ? 2  what are they doing ? walking  are they male or female ? both  are there any animals ? no  are there any trees ? no  are there any animals ? no  are there any animals ? no</p>	<p>how old is the <b>family</b> ? i can't see their faces , i don't know  what are they wearing ? coats and pants  is there a lot of <b>snow</b> ? yes  do you see any <b>trees</b> ? no  what color are the <b>benches</b> ? black  is there a sidewalk ? yes  do you see any <b>cars</b> ? no  is this a <b>park</b> ? no  what is the <b>weather</b> like ? it 's a black and white photo  what is the <b>child</b> wearing ? black jacket and black pants</p>

Figure 5: Selected examples of dialogs on VisDial v1.0 val. Red marks the entity selected at current round.