Object Category-Based Visual Dialog for Effective Question Generation

Feifei Xu, Yingchen Zhou(✉), Zheng Zhong, and Guangzhen Li

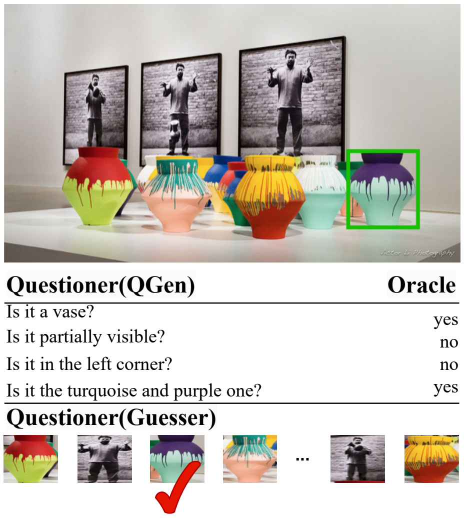
School of Computer Science and Technology, Shanghai University of Electric Power,
Shanghai, China
xufeifei@shiep.edu.cn, zhouyingchen@mail.shiep.edu.cn,
zhongzhengvqa@163.com, liguangzhen@mail.shiep.edu.cn

Abstract. GuessWhat?! is a visual dialog dataset that consists of a series of goal-oriented questions and answers between a questioner and an answerer. The purpose of the task is to enable the questioner to identify the target object in an image based on the dialogue history. A key challenge for the questioner model is to generate informative and strategic questions that can narrow down the search space effectively. However, previous models lack questioning strategies and rely only on the visual features of the objects without considering their category information, which leads to uninformative, redundant or irrelevant questions. To overcome this limitation, we propose an Object-Category based Visual Dialogue (OCVD) model that leverages the category information of objects to generate more diverse and instructive questions. Our model incorporates a category selection module that dynamically updates the category information according to the answers and adopts a linear category-based search strategy. We evaluate our model on the GuessWhat?! dataset and demonstrate its superiority over previous methods in terms of generation quality and dialogue effectiveness.

Keywords: Visual dialog· Question generation· Category information· Questioning strategy.

1 Introduction

In recent years, the domains of vision and language, especially image captioning [31,33,36], vision-and-language navigation [4,8,18], and visual dialog [9,13,32], have attracted increasing attention and research due to the continuous development of artificial intelligence technology and deep learning algorithms. In particular, visual dialog researchers have proposed several different visual dialog tasks, such as VisDial [3,7,9], GuessWhat?! [11,24,29], and GuessWhich [6,20,39], etc. Among these, GuessWhat?! is a goal-oriented visual dialog dataset that involves two players engaged in a question-and-answer session on a single image. Specifically, the Oracle randomly selects a target object from the image, and the Questioner agent asks a series of questions to identify that object while receiving binary answers (i.e., Yes or No) from the Oracle. An example of GuessWhat?! is shown in Figure 1. Generating more efficient questions is what we are aiming for to help the model guess the target object faster. Researchers have divided the questioner agent task into two subtasks, namely the Guesser and the Question Generation (QGen). The Guesser deduces the target object based on dialog history, while the QGen generates relevant questions to aid the Guesser's inference. Independent training of the subtasks allows them to focus on their specific goals, resulting in improved performance. Modeling QGen is crucial for the success of the game, as high-quality questions yield more information about the target object. Furthermore, research on the QGen model facilitates the creation of inferential questions. Most importantly, the QGen model influences the selection of subsequent questions by leveraging previous questions and answers, thereby enhancing the system's conversational reasoning and decision-making capabilities. In this paper, we mainly focus on QGen.



Questioner(QGen)	Oracle
Is it a vase?	yes
Is it partially visible?	no
Is it in the left corner?	no
Is it the turquoise and purple one?	yes

Fig. 1: An Example of GuessWhat?! dataset

The previous QGen models suffer from two major limitations. First, most existing work concerns multimodal fusion [10, 17, 23] and model learning [2, 11, 23, 27, 37], while neglecting effective question generation strategies. Consequently, repetitive and meaningless questions are always generated. To address this problem, researchers employ various strategies to minimize the number of questions. For instance, Testoni et al. [28] propose a visual dialogue strategy that generates more human-like questions, while Shi et al. [35] introduce a sentence-level questioning strategy that generates different types of questions. Notably, [11] indicates that object categories can help humans use linear search strategies to promptly guess the target object. Guiding the question generation under category information can narrow down the search space, allowing Guesser to guess the target object as early as possible. Nonetheless, to the best of our knowledge, no work has been mentioned using category information to guide question generation.

Second, image information is not fully utilized. Existing work encodes either a whole image [11, 15, 22, 25, 27, 28, 38] or the extracted object [5, 20, 24, 29]. If the object's category information is introduced, QGen can generate further fine-grained questions guided by object categories, reducing the occurrence of repeated questions. Faster R-CNN has been shown to be able to detect object category information in images [19]. We can obtain the category information of each object with the help of Faster R-CNN for better question generation.

In this paper, we propose a novel question generation model, Object Category based Visual Dialogue (OCVD). An Object Information Extraction Module is employed to extract the feature and category information of the object. A Category Selection Module is put forward to select the appropriate category in the current round, based on the historical responses. We compute object-category similarity to acquire category-level attention distribution. Together with object-level attention distribution from the Object-Level Attention Update Module, the object features can be updated. Finally, the Object-Self Difference Attention Module is exploited to attain the final visual representation. To make better use of the category information, we connect the final visual representation to the category information to generate the new question. Experimental results demonstrate that our proposed model achieves state-of-the-art performance in the GuessWhat?! task. Additionally, the model introduces new information into the question generation model that helps to generate more informative and strategic questions, thus reducing the search space more efficiently.

Our contributions can be summarized as follows.

- First, we propose a novel question generation model OCVD based on an object category mechanism.
- Second, to simulate human thinking process, a linear search strategy is improved by adjusting the category priority in order to have a higher probability of guessing the target object.
- Third, to enable the model to better obtain useful information from object categories, we design a category selection module that guides the model to generate more informative and valuable questions. For all we know, it is the first time to consider object categories in QGen.
- Finally, we conduct supervised learning and reinforcement learning to train our model and achieve state-of-the-art results in the QGen task.

2 Related Work

Visual dialog is considered as one of the significant research tasks in the domain of vision and language. VisDial [9] and GuessWhat?! [11] are the most common datasets for visual dialog tasks. These datasets involve several rounds of question-answer sessions between two participants, which are based on a single image. Nevertheless, a crucial difference exists between these datasets. In VisDial, the questioner is unable to perceive the image, while the answerer can see the image and answer questions about it. Hence, VisDial models have generally focused on the role of the answerer. Conversely, in GuessWhat?!, both the questioner and the answerer have the same access to the image. The answerer must select an object in the image as a target object and respond to the questioner's inquiry. The questioner's task is to ask questions to identify the object chosen by the answerer. The questioner's role is more intricate than that of the answerer, which involves complex interactions between visual, language, and guessing behaviors, with a higher emphasis on generating goal-oriented questions in visual dialogue tasks.

The QGen model is first introduced by De Vries et al. [11]. It employs an encoder-decoder architecture that encodes the previous round's dialogue using the HRED [21] model's encoder. The result of encoder is connected with the image's VGG features, and both of them are fed into the LSTM [12] to generate questions. Supervised learning is used to train the model by maximizing the conditional log-likelihood. However, the supervised learning framework does not consider the dialogue strategy. To address this issue, Strub et al. [27] proposes a reinforcement learning approach using a policy gradient algorithm to optimize supervised models. The QGen model is optimized by using the supervised trained Oracle and Guesser model to build environment. It uses the final goal as a reward to optimize the question sequence and find the correct object. Zhang et al. [37] proposes a reinforcement learning model that assigns different intermediate rewards to each question to improve the quality of the question and generates concise and informative questions that aid in achieving the final goal. Abbasnejad et al. [2] employs a Bayesian deep learning approach to quantify uncertainty in the internal representation of reinforcement learning models and introduces an information search decoder that accounts for the environment's uncertainty and dialogue history, enabling a more accurate selection of words in each question. Zhao et al. [38] designs a QGen model based on Seq2Seq and introduces the Tempered Policy Gradients method to train the model. They dynamically adjust the temperature of each operation according to the operation frequency of each time step, resulting in better training effect and stability. The questioner task relies on two separate models: QGen and Guesser. However, Shekhar et al. [24] introduces a shared dialogue state encoder that integrates both models, resulting in improved performance and efficiency. This is achieved through a cooperative learning training approach, which tightly combines the two tasks and enables information sharing and interaction. The advantage of this approach is the ability to enhance the interdependence between the models, resulting in more accurate and meaningful questions being generated. Lee et al. [15] proposes an information theoretic algorithm, AQM, grounded in the theory of mind. This approach replaces the training task of question generation with training a neural network to infer answer probabilities. Shukla et al. [25] combines reinforcement learning with regularized information gain to construct a reward function that trains QGen model. This approach is based on the ideology that humans attempt to maximize the expected regularized information gain when asking questions. Shekhar et al. [22] adds a dialogue manager component to the QGen model to determine whether the question generator should continue asking questions or whether the Guesser should guess the target object after each Q&A pair. Pang et al. [17] proposes a Visual Dialogue State Tracking (VDST) approach for question generation. This model tracks the process state of the dialogue, updates the distribution of objects in the image, and adjusts the representation at the end of each round of dialogue, thus guiding the QGen model to ask different meaningful questions. Tu et al. [30] contends that previous models lack shared and a priori knowledge of visual language representation. To overcome this limitation, they leverage the pre-trained visual language model VilBERT to provide improved visual and language representation for dialogue agents. To this end, they propose new Oracle, Guesser, and Questioner models

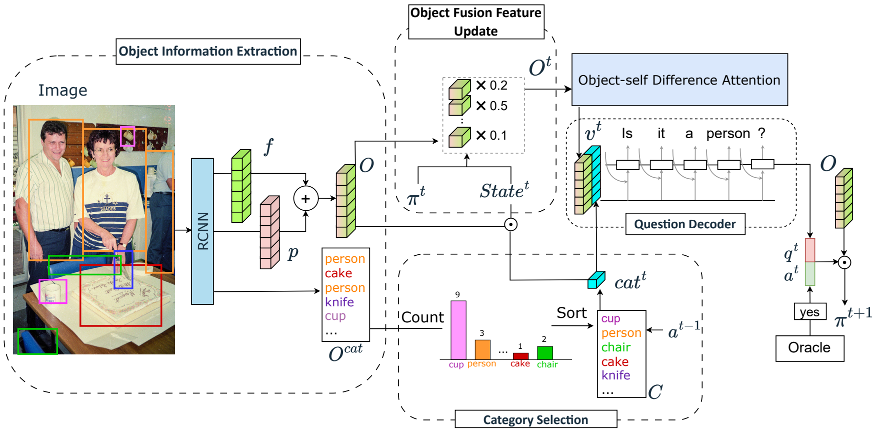


Fig. 2: Overall structure of OCVD model. p is the object position feature. f is the object feature. O is the initial object fusion feature. O^{cat} is the category information. COUNT and SORT are the aggregation and sort operations, respectively. cat^t is the object category to be focused on by the model in round t. O^t is the object fusion feature in round t. π^t and $State^t$ are the object-level attention distribution and category-level attention distribution in round t, respectively. v^t is the final visual representation. a^{t-1} and a^t are the answers of the previous and current rounds, respectively.

employing the pre-trained models. Testoni et al. [28] proposes a beam search reranking strategy called confirm-it, which seeks confirmation information during a dialogue. This approach mirrors human behavior in everyday dialogue, resulting in more natural and human-like questions generated by the model. Shi et al. [35] suggests a questioning strategy based on question categories, where questions are classified into four categories: object, color, location, and other. By analyzing the dialogue history and image features, appropriate question categories are selected to generate targeted questions.

The above methods still generate repetitive or nonsensical questions. In this paper, we use the category information of the object to simulate the way humans ask questions, thus facilitating the QGen model to generate more effective questions.

3 Model

In this section, we will introduce our question generation model OCVD in detail. The dialogue history is represented as $H_{t-1} = \{q^0, a^0, q^1, a^1, \dots, q^{t-1}, a^{t-1}\}$, while the current question $q^t = \{w_1^t, w_2^t, \dots, w_S^t\}$ is a sequence of words with a length of S in round t. The answer $a^t \in \{< \text{Yes} >, < \text{No} >, < \text{NA} >\}$ is restricted to one of three options: yes, no, or not applicable. Additionally, $I \in \mathbb{R}^{H \times W}$ represents the image with a given height H and width W. The model generates the current round question q^t by considering the previous question q^{t-1} and the provided image I. Figure 2 depicts the general structure of the model.

3.1 Object Information Extraction

we apply the Faster-RCNN [19] algorithm to extract an 8-dimensional object position feature p and a d_0-dimensional object feature f from the provided image. Rather than directly concatenating these two vectors, we leverage two fully connected layers to perform projection and normalization on them, respectively. Next, we employ summation and averaging operations to integrate these two features, which yields the object fusion feature O corresponding to the provided image I.

$$y = LayerNorm(W_1 f + b_1), \tag{1}$$

$$x = LayerNorm(W_2 p + b_2), \tag{2}$$

$$O = (x + y)/2. \tag{3}$$

where $W_1 \in \mathbb{R}^{d_0 \times d}$, $W_2 \in \mathbb{R}^{8 \times d}$, $O \in \mathbb{R}^{k \times d}$ including k objects $o \in \mathbb{R}^d$. This treatment aims to balance the incorporation of two distinct types of features and achieve a more integrated feature representation.

Our proposed model departs from the baseline model developed by Pang et al. in which we not only utilize objects and bounding boxes representation vectors but also employ object category information $O^{cat} = [o_1^{cat}, o_2^{cat}, \ldots, o_k^{cat}]$ extracted by the Faster-RCNN object detection model. O^{cat} provides category information for k objects, enabling us to obtain a more comprehensive understanding of the objects in the input image.

3.2 Category Selection

To effectively use this category information, we can analyze the distribution of various object categories in the image. This analysis helps us identify which category is more critical for recognizing the target object. In short, if a category appears more often in an image, then it is more likely to be the category of the target object. Therefore, we process the category information by defining an equation that incorporates it into our model.

$$C = \text{Sort}\left(\text{Counter}\left(\left[o_1^{\text{cat}}, o_2^{cat}, \ldots, o_k^{cat}\right]\right)\right), \tag{4}$$

where Counter is the aggregation function used to determine the number of objects in each category, and Sort is the process of arranging the categories in descending order based on their object counts, ultimately generating a list of candidate categories C. Suppose there are m different categories in the O^{cat}. The set of candidate categories is represented as $C = [c_1, c_2, \ldots, c_m]$.

People often quickly exclude or identify objects based on different object categories. Similar to human thinking, we design a category selection module. This module is designed to select the most relevant category for the question and guide the model's search process accordingly. To achieve this, we begin by setting a Boolean variable called $nofind$ to `True`, which serves as a flag to determine whether there are any candidate categories left to explore. Then, a recursive selection mechanism is employed to update the variables $index$ and $nofind$ based on the answer from the previous round.

$$\begin{cases} index{=}index, nofind{=}\text{False} & \text{if } a^{t-1}{=}\text{ yes} \\ index{=}index{+}1 \quad \text{if } a^{t-1}{=}no \ \& \ nofind{=}\text{True}, \end{cases} \tag{5}$$

where $index$ represents the index of the candidate category list, which is used to update the object category selected in each round of dialogue. The initial value of $index$ is set to 0, indicating that the first element c_1 of the candidate category list is used as the initial category in the first round of dialogue. cat^t denotes the category selected in the tth round of dialogue, and its update is defined by the equation:

$$cat^t = C[index]. \tag{6}$$

3.3 Object Fusion Feature Update

In order to encourage the model to focus more on the object features that are consistent with the selected category cat^t, we compute the similarity score between the object fusion features O and cat^t. Specifically, the similarity score is computed using a similarity function that measures the similarity between the fused feature representation of an object and the feature representation of the selected category.

$$Score^t\left(O, cat^t\right) = \operatorname{softmax}\left(\frac{Ocat^t}{\sqrt{d}}\right), \tag{7}$$

where $O \in \mathbb{R}^{k\times d}$, $cat^t \in \mathbb{R}^{d\times 1}$, and $Sorce^t \in \mathbb{R}^{k\times 1}$. In the VDST model, the cumulative attention distribution on the kth object of the tth round is denoted as $\pi^{(t)} \in \mathbb{R}^{k\times 1}$, and it is updated in each round of the dialogue. To combine the object-level attention distribution $\pi^{(t)}$ with the category-level attention distribution $Score^t$, we introduce the concept of $State^t$. We update the object fusion feature O using $State^t$ and the formula for updating the object representation is as follows:

$$\text{State }^t = \operatorname{softmax}\left(\left(\pi^t + \text{ Score }^t\right)/2\right), \tag{8}$$

$$O^t = \left(\text{ State }^t\right)^T O. \tag{9}$$

3.4 Object-self Difference Attention Module

To obtain the final visual representation, the VDST model uses Object-self Difference Attention to capture the visual differences between objects, the result of which is used as the visual context. Object-self Difference Attention is defined by the following equation:

$$v^t = \operatorname{softmax}\left(\left[o_i^t \odot \left(o_i^t - o_k^t\right)\right] W\right)^T O^t, \tag{10}$$

where $o_i^t, o_k^t \in O^t$. Attention towards objects may change in different rounds, and thus the visual representation v^t in each round can dynamically change under the influence of $State^t$.

3.5 Question Decoder

We choose to use LSTM as the question decoder since it possesses the memory property that can better generate complex natural language questions. To guide the model to generate dialogues related to the selected object category cat^t, we additionally incorporate cat^t into the input of the LSTM.

$$w_{i+1}^t = \mathrm{LSTM}\left(\left[v^t; cat^t; w_i^t\right]\right), \tag{11}$$

where w_i^t denotes the ith word in the question q^t, while $[;]$ indicates concatenation. The final hidden state of the LSTM decoder serves as the representation of the question q^t.

3.6 Object-Level Attention Update

After obtaining the answer a^t from the Oracle, we concatenate the embedding of a^t with the representation of q^t, resulting in $h^t = [q^t; a^t]$. Based on the question-answer pair and object representation, we update π^t.

$$\pi^{\mathrm{t}+1} = \mathrm{Norm}\left(\mathrm{softmax}\left(\frac{\tanh\left(O^{\mathrm{t}}U^T \odot V^T h^{\mathrm{t}}\right)}{\sqrt{d}}\right)\pi^{\mathrm{t}}\right). \tag{12}$$

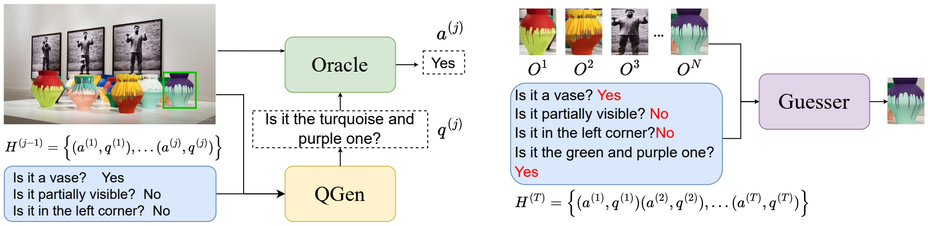
4 Experiments

4.1 Dataset

To evaluate our model, we use the GuessWhat?! dataset, which consists of 66k images and 821k question-answer pairs in 155k dialogues (Each image may correspond to multiple dialogues). The dialogues are about a target object in the image that Guesser tries to guess by asking yes/no questions to Oracle. The task is considered successful if the guesser correctly identifies the object. Following previous work, we split the dataset into training, validation, and test sets with a ratio of 70%, 15%, and 15%, respectively. We only include the dialogues that are successful (84.6% of the total) for training and evaluation, and exclude those that are unsuccessful (8.4%) or incomplete (7.0%).

4.2 Evaluation Metrics

Task success rate: The QGen model's performance is measured by the probability that the Questioner in a game can successfully identify the target object within a certain number of rounds. This means that the Questioner has to use


Oracle
$a^{(j)}$
Yes
$q^{(j)}$
QGen
$H^{(j-1)} = \left\{(a^{(1)}, q^{(1)}), \dots (a^{(j)}, q^{(j)})\right\}$
O^1
O^2
O^3
...
O^N
Guesser
$H^{(T)} = \left\{(a^{(1)}, q^{(1)})(a^{(2)}, q^{(2)}), \dots (a^{(T)}, q^{(T)})\right\}$

Is it the turquoise and purple one?

Is it a vase? Yes
Is it partially visible? No
Is it in the left corner? No

Is it a vase? Yes
Is it partially visible? No
Is it in the left corner?No
Is it the green and purple one? Yes

Fig. 3: An Example of Qgen, Oracle, and Guesser. On the left side, Oracle and QGen generate questions and answers through interaction. On the right side, Guesser makes a guess based on the complete dialogue history when a predefined number of rounds T is reached.

relevant questions to achieve this objective. Nonetheless, the QGen model is constrained by the Guesser and Oracle models, so it can only be evaluated by associating with Guesser and Oracle models. A game example of the Qgen, Oracle, and Guesser models is shown in Figure 3.

Rate of Games with Repeated Questions: Repeated questions are undesirable in a dialog, as they can reduce the task success rate. A question is considered repetitive if it has already been asked in the history of a dialogue. Rate of Games with Repeated Questions is the ratio of the number of games containing repeated questions to the total number of games. This metric reflects the validity and diversity of the questions generated by the model.

4.3 Experiment Settings

We use Faster-RCNN to extract a feature vector of dimension 1024 for each image region. We select $k = 36$ objects from each image based on object detection. Both historical questions and answers are embedded in 512 dimensions each, and the dimension of category information embedding is also 512. Therefore, the LSTM hidden unit number is 512.

We implement our model using PyTorch and train it in two stages: supervised learning (SL) and reinforcement learning (RL). In the SL stage, we use the Adam [14] optimizer with a learning rate of 1e-4 and a batch size of 64. We train the Guesser and Oracle models for 30 epochs each and train the QGen model for 50 epochs. In the RL stage, we follow the same setup as de Vries et al. [27] and train the QGen model for 100 epochs using stochastic gradient descent (SGD) with a learning rate of 1e-4 and a batch size of 64.

4.4 Results

Game success rate: We evaluate our model on the game success rate and compare it with several recent state-of-the-art models in this field: SL [11], GDSE [24], RL [31], TPG [38], VQG [37], ISM [1], Bayesian [2], RIG [25], VDST [17], CSQG [35], VilBert-Questioner [30], ISM [1], and ADVSE-QGen [34]. Table 1 and Table 2 show the results of the comparison under two settings: New Object,

Table 1: Task success rate for SL models. The results of the baseline models are from its original paper.

NewObject			
	Max turn	Greedy	BSearch
SL [11]	5	43.5	47.1
VDST-SL [17]	5	49.49	–
	8	48.01	–
ADVSE-QGen [34]	5	50.66	47.47
TPG-SL [38]	8	48.77	–
CSQG [35]	5	53.2	52.4
	8	54.4	53.9
ours	5	**55.3**	**53.7**
	8	**55.9**	**54.8**
NewGame			
SL [11]	5	40.8	44.6
	8	40.7	–
VDST-SL [17]	5	45.94	–
	8	45.03	–
ADVSE-QGen [34]	5	47.03	44.7
CSQG [35]	5	49.9	48.1
	8	51.7	49.7
GDSE-SL [24]	5	47.8	–
	8	49.7	–
VilBert-Questioner [30]	–	52.5	–
ours	5	**52.6**	**50.1**
	8	**53.3**	**51.5**

Table 2: Task success rate for RL models. The results of the baseline models are from its original paper.

New Object			
	Max turn	Greedy	BSearch
RL [31]	5	60.3	60.2
	8	58.2	53.9
VQG [37]	5	63.6	63.9
ISM [1]	–	64.2	–
Bayesian [2]	5	62.1	63.6
RIG as rewards [25]	8	63	63.08
RIG(0-1 rewards) [25]	8	63.19	62.57
VDST [17]	5	67.07	67.81
	8	70.55	71.03
ours	5	**69.3**	**68.9**
	8	**71.8**	**72.1**
New Game			
RL [31]	5	40.8	44.6
	8	40.7	–
VQG [37]	5	60.7	60.8
ISM [1]	–	62.1	–
RIG as rewards [25]	5	59.8	60.6
	8	59	60.21
RIG(0-1 rewards) [25]	8	61.18	59.79
VDST [17]	5	64.36	64.44
	8	67.73	67.52
GDSE-CL [24]	5	53.7	–
	8	58.4	–
ours	5	**66.4**	**65.9**
	8	**67.9**	66.7

where we use images from the training set but change the target object; and New Game, where we use data from the test set with both images and targets being new. We conduct the experiments with 5 and 8 rounds of dialogues based on beam search or greedy search, respectively.

It is important to note that the focus of recent research has shifted to building Guesser vs. Oracle models rather than QGen model improvements. Due to this trend, no newer QGen models are available for comparison. Nevertheless, our work is dedicated to improving and optimizing the performance of QGen models and demonstrating the effectiveness of our proposed approach by comparing it with classical methods.

We compare our model with several existing models and report the results in Table 1 and Table 2. We categorize the models into SL models and RL models. Table 1 shows that our model achieves the highest success rate among the SL models. It can be seen that our model not only outperforms the other supervised learning models but also outperforms the QGen model that uses a pre-trained visual language encoder (VilBert-Questioner). Our model reaches a success rate of 55.9% on New Object and 53.3% on New Game, establishing new state-of-the-art results with SL. Table 2 shows that our model also exceeds the RL models,

Table 3: Rate of games with repeated questions of different questioner models. OCVD-SL refers to OCVD models trained using supervised learning only.

% Games with Repeated Q's	
SL [11]	93.5
RL [31]	96.47
GDSE-SL [24]	55.8
GDSE-CL [24]	52.19
VDST-SL [17]	40.05
VilBERT-Questioner [34]	32.56
OCVD-SL	**31.85**
VDST [17]	21.9
ours	**18.73**
Human	N/A

Table 4: Experimental results of ablation studies.

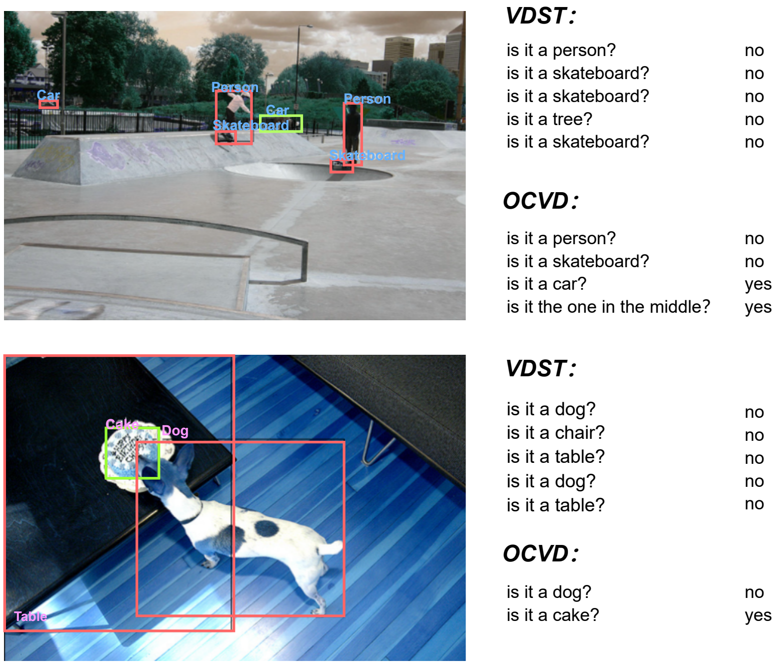
Model	New game
OCVD(full model)	66.4
w/o similarity score	66.0
w/o Category selection module	64.9
w/o Category information	64.36

achieving a 72.1% success rate on New Object and a 67.9% success rate on New Game, which indicates the effectiveness of OCVD.

Repeated questions: Besides the task success rate, another important aspect of evaluating QGen model is the quality of the generated question, such as its relevance and informativeness and the avoidance of redundancy. To measure the quality of questions, we use the rate of games with repeated questions. Table 3 shows the rate of games with repeated questions for different models. From the results, it can be easily found that our OCVD has the lowest rate of games with repeated questions.

Qualitative results: Figure 4 shows some dialogue samples generated by our OCVD model and a baseline VDST model. We can see that our model can effectively generate relevant questions based on the object categories in the image. Furthermore, the object categories can be dynamically updated according to the answers. For instance, in the first example, the image contains three possible object categories: "people, car, skateboard". The model first asks if the target object is a person. When the answer is "no", it switches to another object category until the answer is "yes". After determining the target object category, it will ask more specific questions to identify which car it is. In comparison, under the same training settings, the VDST model fails to learn this questioning strategy and generates unnatural dialogues. As illustrated in Figure 5, we visualize the learned attention graph. The regions enclosed in red boxes signify higher attention weights, and we depict the bounding boxes corresponding to the first five highest scores.

Ablation Studies: To verify the effectiveness of our proposed Object Category based Visual Dialogue (OCVD) model, we conduct a series of ablation experiments on the GuessWhat?! dataset. We use the greedy search and maximum of 5 questions on the test set and separately investigate different components of our model. The ablation experiment result shows in Table 4.



VDST:

is it a person?	no
is it a skateboard?	no
is it a skateboard?	no
is it a tree?	no
is it a skateboard?	no

OCVD:

is it a person?	no
is it a skateboard?	no
is it a car?	yes
is it the one in the middle?	yes

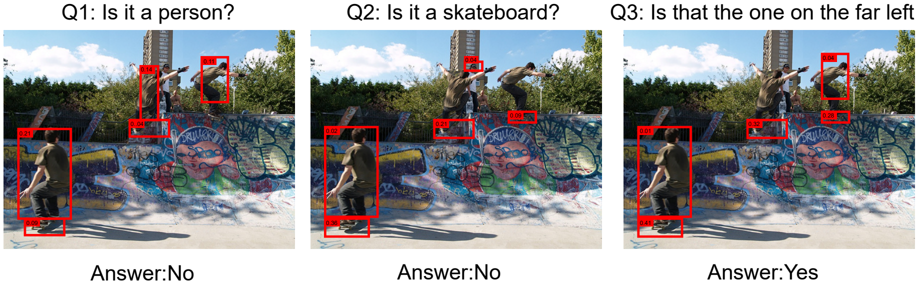
VDST:

is it a dog?	no
is it a chair?	no
is it a table?	no
is it a dog?	no
is it a table?	no

OCVD:

is it a dog?	no
is it a cake?	yes

Fig. 4: Game examples of guessing the target object. The target object is highlighted in the green box.


Q1: Is it a person?

Q2: Is it a skateboard?

Q3: Is that the one on the far left

Answer:No

Answer:No

Answer:Yes

Fig. 5: Learned Attention Visualization with Top Five Highest Scores

First, the component that calculates the similarity score between visual features and the selected category is removed. This component guides the model to concentrate on object features of the same category. The results show that without this component, the task success rate of our model decreases by 0.4% without the similarity score calculation. This indicates that this component is effective in guiding the model to generate category-related questions by focusing on the relevant object features.

Second, the category selection module is omitted, which dynamically adjusts the category information based on the answers to implement a linear search questioning strategy. The results show that randomly selecting category information instead of dynamically adjusting it according to the answers results in a 1.1% decrease in our model's task success rate. Therefore, we conclude that the category selection module plays an important role in improving the model's performance.

Finally, we delete the category information part, which helps the model with additional object category information to generate more targeted questions. The results show that removing the category information decreases the task success rate of the model to 64.36%, which is lower than the task success rate of the full model. Therefore, we argue that the consideration of category information is valid for generating more effective questions. Overall, the ablation experiments demonstrate the effectiveness of our proposed OCVD model in generating more diverse and effective questions by utilizing the object category information.

5 Conclusions

In this paper, we propose a novel question generation model, Object Category based Visual Dialogue (OCVD), which uses object category information as a clue for generating valid questions. This approach aims to help the model identify the target object by its category. We train the OCVD model on the GuessWhat?! dataset and the results show that it can implement an effective linear search strategy based on object categories. In addition, the model performs well on the GuessWhat?! task, which indicates that using object category information is effective for question generation. Ablation experiments further confirm that object category information plays a vital role in enhancing the OCVD model performance. Specifically, when object category information is removed, the performance of the model significantly decreases, and the quality of the generated dialogues is substantially reduced. The OCVD model exhibits promise for extensibility across diverse visual language tasks. Specifically, within educational contexts, the OCVD model can enhance learning by generating image-related questions. These questions serve to assess students' understanding of visual content and encourage in-depth exploration of educational material. We argue that the OCVD model's utility extends beyond the GuessWhat?! visual dialog task, making it applicable to a broader range of visual dialog tasks, including generating datasets for conversations about images, highlighting the model's potential in supporting various aspects of visual learning. Nevertheless, it should be noted that this study only considers the category information of objects in images and does not incorporate other important attribute information such as color, shape, and location. Thus, in the future, we will incorporate multiple attribute information to generate more effective and natural questions. Furthermore, we will explore optimization methods for question generation strategies to enhance the performance of the model. Furthermore, we will explore ways to optimize the GuessWhat?! dataset [16, 26] to improve the performance of the model.

References

1. Abbasnejad, E., Wu, Q., Abbasnejad, I., Shi, J.Q., van den Hengel, A.: An active information seeking model for goal-oriented vision-and-language tasks. ArXiv **abs/1812.06398** (2018)
2. Abbasnejad, E., Wu, Q., Shi, J.Q., van den Hengel, A.: What's to know? uncertainty as a guide to asking goal-oriented questions. 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) pp. 4150–4159 (2018)
3. Agarwal, S., Bui, T., Lee, J.Y., Konstas, I., Rieser, V.: History for visual dialog: Do we really need it? In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. pp. 8182–8197 (2020)
4. Anderson, P., Wu, Q., Teney, D., Bruce, J., Johnson, M., Sünderhauf, N., Reid, I., Gould, S., Van Den Hengel, A.: Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3674–3683 (2018)
5. Bani, G., Belli, D., Dagan, G., Geenen, A., Skliar, A., Venkatesh, A., Baumgärtner, T., Bruni, E., Fernández, R.: Adding object detection skills to visual dialogue agents. In: ECCV Workshops (2018)
6. Chattopadhyay, P., Yadav, D., Prabhu, V., Chandrasekaran, A., Das, A., Lee, S., Batra, D., Parikh, D.: Evaluating visual conversational agents via cooperative human-ai games. In: Proceedings of the AAAI Conference on Human Computation and Crowdsourcing. vol. 5, pp. 2–10 (2017)
7. Chen, C., Tan, Z., Cheng, Q., Jiang, X., Liu, Q., Zhu, Y., Gu, X.: Utc: a unified transformer with inter-task contrastive learning for visual dialog. In: Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition. pp. 18103–18112 (2022)
8. Chen, S., Guhur, P.L., Tapaswi, M., Schmid, C., Laptev, I.: Think global, act local: Dual-scale graph transformer for vision-and-language navigation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 16537–16547 (2022)
9. Das, A., Kottur, S., Gupta, K., Singh, A., Yadav, D., Moura, J.M., Parikh, D., Batra, D.: Visual dialog. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 326–335 (2017)
10. Das, A., Kottur, S., Moura, J.M.F., Lee, S., Batra, D.: Learning cooperative visual dialog agents with deep reinforcement learning. 2017 IEEE International Conference on Computer Vision (ICCV) pp. 2970–2979 (2017)
11. De Vries, H., Strub, F., Chandar, S., Pietquin, O., Larochelle, H., Courville, A.: Guesswhat?! visual object discovery through multi-modal dialogue. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 5503–5512 (2017)
12. Gers, F.A., Schmidhuber, J., Cummins, F.: Learning to forget: Continual prediction with lstm. Neural Computation **12**, 2451–2471 (2000)
13. Guo, D., Wang, H., Zhang, H., Zha, Z.J., Wang, M.: Iterative context-aware graph inference for visual dialog. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 10055–10064 (2020)
14. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. CoRR **abs/1412.6980** (2014)
15. Lee, S.W., Heo, Y.J., Zhang, B.T.: Answerer in questioner's mind: Information theoretic approach to goal-oriented visual dialog. In: Neural Information Processing Systems (2018)

16. Oshima, R., Shinagawa, S., Tsunashima, H., Feng, Q., Morishima, S.: Pointing out human answer mistakes in a goal-oriented visual dialogue. arXiv preprint arXiv:2309.10375 (2023)
17. Pang, W., Wang, X.: Visual dialogue state tracking for question generation. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 34, pp. 11831–11838 (2020)
18. Pashevich, A., Schmid, C., Sun, C.: Episodic transformer for vision-and-language navigation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 15942–15952 (2021)
19. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. Advances in neural information processing systems **28** (2015)
20. Sang-Woo, L., Tong, G., Sohee, Y., Jaejun, Y., Jung-Woo, H.: Large-scale answerer in questioner's mind for visual dialog question generation. In: Proceedings of International Conference on Learning Representations, ICLR (2019)
21. Serban, I., Sordoni, A., Bengio, Y., Courville, A.C., Pineau, J.: Hierarchical neural network generative models for movie dialogues. ArXiv **abs/1507.04808** (2015)
22. Shekhar, R., Baumgärtner, T., Venkatesh, A., Bruni, E., Bernardi, R., Fernández, R.: Ask no more: Deciding when to guess in referential visual dialogue. In: Proceedings of the 27th International Conference on Computational Linguistics. pp. 1218–1233 (2019)
23. Shekhar, R., Venkatesh, A., Baumgärtner, T., Bruni, E., Plank, B., Bernardi, R., Fernández, R.: Beyond task success: A closer look at jointly learning to see, ask, and guesswhat. In: North American Chapter of the Association for Computational Linguistics (2018)
24. Shekhar, R., Venkatesh, A., Baumgärtner, T., Bruni, E., Plank, B., Bernardi, R., Fernández, R.: Beyond task success: A closer look at jointly learning to see, ask, and guesswhat. In: Proceedings of NAACL-HLT. pp. 2578–2587 (2019)
25. Shukla, P., Elmadjian, C., Sharan, R., Kulkarni, V., Turk, M., Wang, W.Y.: 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. pp. 6442–6451 (2020)
26. Sicilia, A., Alikhani, M.: Learning to generate equitable text in dialogue from biased training data. arXiv preprint arXiv:2307.04303 (2023)
27. Strub, F., de Vries, H., Mary, J., Piot, B., Courville, A., Pietquin, O.: End-to-end optimization of goal-driven and visually grounded dialogue systems harm de vries. In: International Joint Conference on Artificial Intelligence (2017)
28. Testoni, A., Bernardi, R.: Looking for confirmations: An effective and human-like visual dialogue strategy. In: Conference on Empirical Methods in Natural Language Processing (2021)
29. Testoni, A., Bernardi, R.: Garbage in, flowers out: Noisy training data help generative models at test time. IJCoL. Italian Journal of Computational Linguistics **8**(8-1) (2022)
30. Tu, T., Ping, Q., Thattai, G., Tur, G., Natarajan, P.: Learning better visual dialog agents with pretrained visual-linguistic representation. 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) pp. 5618–5627 (2021)
31. Wang, Y., Xu, J., Sun, Y.: End-to-end transformer based model for image captioning. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 36, pp. 2585–2594 (2022)

32. Wang, Y., Joty, S., Lyu, M., King, I., Xiong, C., Hoi, S.C.: Vd-bert: A unified vision and dialog transformer with bert. In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP). pp. 3325–3338 (2020)
33. Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A., Salakhudinov, R., Zemel, R., Bengio, Y.: Show, attend and tell: Neural image caption generation with visual attention. In: International conference on machine learning. pp. 2048–2057. PMLR (2015)
34. Xu, Z., Feng, F., Wang, X., Yang, Y., Jiang, H., Ouyang, Z.: Answer-driven visual state estimator for goal-oriented visual dialogue. Proceedings of the 28th ACM International Conference on Multimedia (2020)
35. Yanan, S., Yanxin, T., Fangxiang, F., Chunping, Z., Xiaojie, W.: Category-based strategy-driven question generator for visual dialogue. In: Proceedings of the 20th Chinese National Conference on Computational Linguistics. pp. 1000–1011 (2022)
36. Yuan, Z., Yan, X., Liao, Y., Guo, Y., Li, G., Cui, S., Li, Z.: X-trans2cap: Cross-modal knowledge transfer using transformer for 3d dense captioning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8563–8573 (2022)
37. Zhang, J., Wu, Q., Shen, C., Zhang, J., Lu, J., van den Hengel, A.: Asking the difficult questions: Goal-oriented visual question generation via intermediate rewards. In: European Conference on Computer Vision (2017)
38. Zhao, R., Tresp, V.: Improving goal-oriented visual dialog agents via advanced recurrent nets with tempered policy gradient. In: LaCATODA@ IJCAI. pp. 1–7 (2018)
39. Zheng, D., Xu, Z., Meng, F., Wang, X., Wang, J., Zhou, J.: Enhancing visual dialog questioner with entity-based strategy learning and augmented guesser. In: Findings of the Association for Computational Linguistics: EMNLP 2021. pp. 1839–1851 (2021)