Learning to Ask Informative Sub-Questions for Visual Question Answering

Kohei Uehara
The University of Tokyo
Tokyo, Japan
uehara@mi.t.u-tokyo.ac.jp

Nan Duan
Microsoft Research Asia
Beijing, China
nanduan@microsoft.com

Tatsuya Harada
The University of Tokyo / RIKEN
Tokyo, Japan
harada@mi.t.u-tokyo.ac.jp

Abstract

VQA (Visual Question Answering) model tends to make incorrect inferences for questions that require reasoning over world knowledge. Recent study has shown that training VQA models with questions that provide lower-level perceptual information along with reasoning questions improves performance. Inspired by this, we propose a novel VQA model that generates questions to actively obtain auxiliary perceptual information useful for correct reasoning. Our model consists of a VQA model for answering questions, a Visual Question Generation (VQG) model for generating questions, and an Info-score model for estimating the amount of information the generated questions contain, which is useful in answering the original question. We train the VQG model to maximize the "informativeness" provided by the Info-score model to generate questions that contain as much information as possible, about the answer to the original question. Our experiments show that by inputting the generated questions and their answers as additional information to the VQA model, it can indeed predict the answer more correctly than the baseline model.

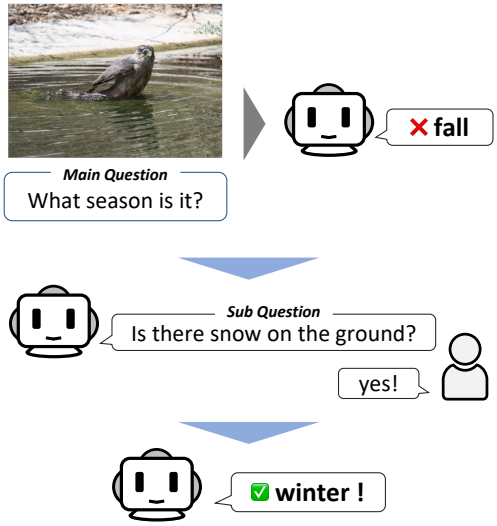


Figure 1. An illustration of our proposed "asking the informative sub-question" framework. When only the main reasoning question like "What season is it?" is input to a VQA model, the model may make a wrong prediction. Even in such a case, our proposed system generates a sub-question related to the main-question and acquires information to make correct reasoning.

1. Introduction

Visual Question Answering (VQA) [3, 11] is a task of answering questions about an image, and is considered to be a crucial task for evaluating the semantic image comprehension level of an image recognition model. Generally, VQA models are designed such that given an image and a question about the image, the model outputs a plausible answer from these two inputs [3, 7, 12, 35].

Unfortunately, the performance of the VQA models is not perfect, and the model often makes incorrect predictions. In particular, it has been pointed out that for questions that require reasoning over world knowledge, the VQA model tends to answer incorrectly or for the wrong reasons [22, 29]. For example (Figure 1), to answer the question "What season is it?", the VQA model needs a reasoning process that "there is snow on the ground in the image, so it is winter." However, it is difficult to train a VQA model to acquire such a reasoning process.

In order to have the model acquire such reasoning ability, [29] proposed a method that utilizes not only a reasoning questions but also Perception questions. Perception questions are questions only require visual perception ability, such as "Is there snow on the ground?" in Figure 1. They constructed a novel dataset that contains Perception questions associated with reasoning questions and train the model to answer both types of the questions simultaneously.

Inspired by their idea, we study the method of asking the perceptual questions that are useful for the model to make correct reasoning. Specifically, our Visual Question Generation (VQG) model generates a "sub-question" about an image and a reasoning question about the image (refer to as "main-question"). By obtaining answers to the generated

sub-questions from someone else, the model can obtain the information necessary to make correct reasoning.

For example, when the main reasoning question is "What season is it?" and the VQA model initially predicts the answer as "fall" (Figure. 1), the VQG model generates a sub-question ("Is there snow on the ground?") that helps the VQA model predict the answer to the main-question. If the model can obtain an answer to the sub-question ("yes"), the VQA model would be able to make a correct prediction; "winter", for the main-question using the sub-question and the answer as an additinal information.

Our proposed pipeline consists of three components: Target VQA model, Info-score model, and VQG model. First, we design the Target VQA model so that it can receive both types of input: main-question only (main-only), and main-question and sub-question (main+sub). When the input is main-only, the Target VQA model predicts the main answer to the main-question, as in the standard VQA model. On the other hand, when the input is main+sub, the Target VQA model predicts the main answer using the sub-question and its answer as additional information in addition to the main-question. Then, the Info-score model predicts the informativeness of the sub-question. Here, we introduce the concept of **info score** to quantify the amount of useful information the sub-question provides to answer the main-question. The info-score is defined as the difference between the loss value when the input to the Target VQA model is "main-only" and the loss value when the input is "main+sub." Finally, we train the VQG model to generate a sub-question from the main-question. To generate a sub-question that can obtain as much information as possible, we train the VQG model by using reinforcement learning with info-score as a reward.

Our contribution is summarized as follows:

1. We propose a novel active VQA method, which generates perceptual sub-questions and obtains their answers from others to serve as auxiliary information to the main-question.

2. We propose "info-score" that measures the amount of information about the main-question can be obtained by the generated sub-question, and train a VQG model using it as a reward.

3. We evaluate the effectiveness of the proposed method by checking its performance when sub-questions generated by the VQG model are input to the Target VQA model.

2. Related Work

2.1. Reasoning Ability of VQA Model

In VQA, the model is supposed to make reasoning based on the input image and question to answer the question. However, it has been pointed out that the model might not learn the inference process based on the image and question content, but capture the bias of the image and language [1, 2, 9, 14, 39]. In order to solve this problem, researchers have focused on the consistency of model responses to similar inputs (e.g., perturbed inputs [1, 9], or rephrased questions [10, 14, 30], perceptual sub-questions [29]). Our research is motivated by [29], which is the study of adding perceptual sub-questions to the reasoning questions. Perceptual sub-questions are defined as the questions that can be answered from visual content of the image. Thus, it is expected that we can obtain a model that can correctly perform visual reasoning by training the model to correctly answer such sub-questions. In their method, the model utilizes human-annotated sub-questions, for which a suitable sub-question must exist in the dataset. We tackle the problem by making the model capable of dynamically generating useful sub-questions.

2.2. VQG as Data Augmentation

There are several studies that used VQG methods to augment the data for VQA. In these studies, the methods for obtaining new questions can be roughly grouped into three categories: applying rule-based operations (e.g., word replacement) to existing questions [5,9,13,37], applying paraphrasing models such as back-translation models to existing questions [14, 30, 34], and directly learning VQG models that generate questions from images [24, 36]. Some studies [24, 31] have tried to perform efficient data augmentation with VQG by incorporating active learning that takes into account the uncertainty of the prediction.

However, the major difference between our method and existing studies is when to use the augmented data. In the existing studies, newly generated questions were added to the training data to be used in the training phase of the VQA model. In this case, for the VQA model to benefit from the questions generated by the VQG model, it is necessary to generate a considerable amount of questions in advance and train the VQA model with them, which is a time-consuming task. In our method, however, the generated questions are used only during the inference phase of the VQA model. Therefore, the performance improvement can be obtained immediately without additional training of the VQA model.

2.3. VQA with Additional Information

Several studies have aimed to improve the performance of VQA models by adding additional information (e.g., scene graphs [6, 8, 32] or knowledge databases [23, 25, 26]) to the input in addition to the VQA questions.

Another effort that is similar to this work used image captions as additional textual information about the image. Some studies in this stream include those that use a caption generator trained separately from the VQA

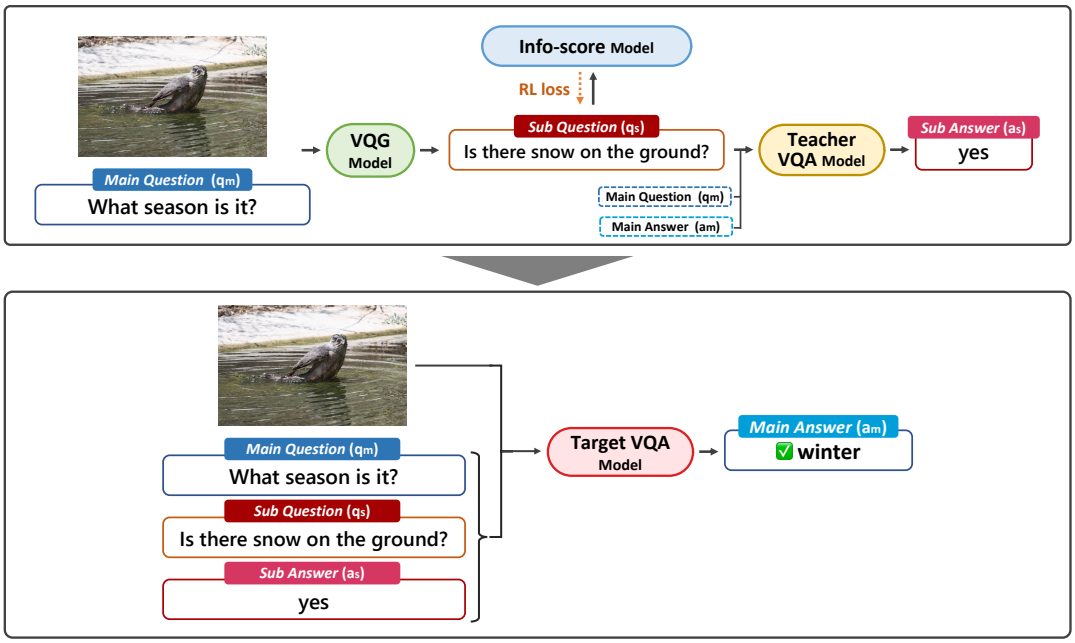


Figure 2. The overall framework of our model. The VQG model generates a sub-question from an image and a main-question. Since we aim to generate questions that contain as much information about the main-question as possible, the VQG model is trained by reinforcement learning with info-score as the reward. The Target VQA model takes the generated sub-question and sub-answer as input in addition to the main-question, and predicts the main-answer by using the added information.

model [15, 18, 19], and those that jointly train VQA and caption generation [38]. However, in order to obtain the information required for a question in such a study, the scene graph generating model or the caption generating model must already know the information that contains the answer to the question. In this study, we use the strategy of "asking additional questions to others" to obtain additional information. Therefore, our approach differs from the above studies in that it is not necessary to prepare a model that already knows the information of the answer to the question.

3. Methods

The overall model is shown in Figure. 2. We refer to the two types of questions used in this study as follows: q_m ("**main-question**") is the reasoning question that is initially asked to the VQA model, and q_s ("**sub-question**") is the perceptual question that is expected to give additional information to that question. Our goal is to use the VQG model to generate a sub-question that contains more helpful information as possible, for answering the main-question. Here, we use the term "**informativeness**" to mean the amount of information the sub-question and its answer a_s (refer as "**sub-answer**") contain to estimate the answer to the main-question.

Our model consists of the following three parts: (1) Target VQA model $\mathcal{A}$, (2) Info-score model $\mathcal{I}$, and (3) VQG model $\mathcal{G}$. $\mathcal{A}$ is the VQA model whose performance is to be improved in this study. We define "**info-score**" I as a metric to quantify informativeness, that is, how close the prediction made by the target VQA model is to the correct answer, when the sub-question and sub-answer is additionally input to the Target VQA model. $\mathcal{I}$ is a regression model that predicts the info-score when a certain "sub-question" is input to a certain "main-question". Then, $\mathcal{G}$ generates a sub-question that will improve the info-score as much as possible. We optimize $\mathcal{G}$ using reinforcement learning loss with I as a reward to maximize the informativeness obtained from the generated questions.

In the inference phase, we input the sub-question generated by the VQG model and sub-answer to the Target VQA model, in addition to the image and main-question. Ideally, the sub-answer should be provided by humans, but in this study, we have another VQA model called Teacher VQA model to provide sub-answer instead of humans.

3.1. Target VQA Model

The Target VQA model $\mathcal{A}$ predict the answer for the main-question, a_m (referred to as "**main-answer**"), i.e., $a_m = \mathcal{A}(v, q)$. Here, v is the input image, q is the input text to the model. This model normally takes as input an image and a question about it, predicts the answer to the question, like the standard VQA model. In this study, the input to the Target VQA model could be one of two types: (A) main only, or (B) main + sub. In case (A), the input

text q is the main-question q_m. While in case (B), q is the conjunction of the main-question, the sub-question, and the sub-answer.

$$q = \begin{cases} q_{\texttt{main}} = q_m & \text{(A) (main only)} \\ q_{\texttt{main+sub}} = [q_m;\ q_s;\ a_s] & \text{(B) (main + sub)} \end{cases} \quad (1)$$

Note that the Target VQA model is trained to predict the answer to the main-question a_m in both cases.

In order to build such a model, we have designed a VQA model based on UNITER [7], a multi-modal Transformer model that has demonstrated high performance in various vision and language tasks. We compute the probablity of the main-answer by using a two-layer MLP on top of cross-modality feature h obtained by UNITER:

$$h = \mathrm{Enc}(v,\ q) \quad (2)$$

$$P_{\mathcal{A}}(a_m \,|\, v,\ q) = \sigma(\mathrm{MLP}(h)) \quad (3)$$

where Enc is a UNITER encoder and σ is a sigmoid function.

The original UNITER encoder was designed to accept only a single question as the input. We make a modification to the input format such that sub-questions and sub-answers can also be used as inputs. Specifically, we add special tokens indicating the type of input texts ([MAIN Q], [SUB Q], and [SUB ANS], respectively) at the beginning and concatenate all of them as necessary. That is, in the case of "main-only", the input text to the model is written as $\{$[MAIN Q], $w_1^{q_m}$, $w_2^{q_m}$, ...$\}$. On the other hand, in the case of "main+sub", the input text to the model is written as $\{$[MAIN Q], $w_1^{q_m}$, ..., [SUB Q], $w_1^{q_s}$, ..., [SUB ANS], $w_1^{a_s}$, ...$\}$. Here, $w_i^{q_m}$, $w_i^{q_s}$, $w_i^{a_s}$ denotes the i-th word of the main-question, sub-question, and sub-answer, respectively. Our loss function for the Target VQA model is a binary cross-entropy loss with soft target scores [7, 35], following the original UNITER VQA model.

$$\begin{aligned} L_{\mathcal{A}}(v,\ q) = &- a_m \log P_{\mathcal{A}}(a_m \,|\, v,\ q) \\ &- (1 - a_m) \log(1 - P_{\mathcal{A}}(a_m \,|\, v,\ q)) \end{aligned} \quad (4)$$

3.2. Info-score Model

The Info-score model $\mathcal{I}$ uses an image v, a main-question q_m, and a sub-question q_s as inputs, and predicts the info-score I, i.e., $I = \mathcal{I}(v,\ q_m,\ q_s)$.

In order to train this model, we first calculate the ground-truth info-score for a pre-trained Target VQA model. The info-score is a quantitative measure of how close to the correct answer the Target VQA model is able to output in the "main + sub" case, compared to the "main-only" case. Specifically, info-score is defined as the difference between the output of the loss function when the input is "main-only" and when the input is "main+sub". Given q_m, q_s, and a_s, the ground-truth info-score is calculated as follows:

$$I_{\mathrm{GT}} = L_{\mathcal{A}}(v,\ q_m) - L_{\mathcal{A}}(v,\ [q_m;\ q_s;\ a_s]) \quad (5)$$

where $L_{\mathcal{A}}$ is the loss function for the Target VQA model. Since the loss value becomes smaller as the output of the model comes closer to the distribution of the correct answer, a larger value of info-score indicates that the sub-question contains more information.

Then, we train the Info-score model to predict the info-score from the image, main-question, and sub-question. This model is used during the training of the VQG model to estimate the info-score to be used as a reward to train the VQG model. The Info-score model is a combination of a multimodal encoder based on UNITER and a head designed to perform regression of the info-score. Following some existing studies on NLVR2 [33], which is a binary classification task for multi-modal inputs, we use the *pair* method [16, 20, 40] to encode the image, the main-question, and the sub-question. In this method, each pair of (image, main-question) and (image, sub-question) are fed into the encoder, and the decoder takes the concatenation of the output of the encoder to make a prediction. The regression head of the decoder uses two layers of MLPs and a sigmoid function as the activation function in the final layer:

$$h_m = \mathrm{Enc}(v,\ q_m) \quad (6)$$

$$h_s = \mathrm{Enc}(v,\ q_s) \quad (7)$$

$$\hat{I} = \sigma(\mathrm{MLP}([h_m;\ h_s])) \quad (8)$$

where h_m and h_s are fused feature for the image and main-question and the image and the sub-question, respectively. Note that we normalized the info-score of the training data so that the maximum value was 1 and the minimum value was 0. The model is trained by minimizing the binary cross entropy loss.

3.3. VQG Model

The VQG model $\mathcal{G}$ is an encoder-decoder model that uses an image v and a main-question q_m as input and generates a sub-question q_s, i.e., $q_s = \mathcal{G}(v,\ q_m)$. The encoder, as in the previous models, uses the UNITER encoder to encode multimodal context information of images and texts. The decoder is designed based on the decoder of BART [17], which is a text-generation model using a Transformer. The BART decoder consists of several blocks of Transformers, each of which is composed of a self-attention and a cross-attention layer. In the early phase of training, the model is trained in a teacher-forcing manner using sub-questions of the training data. Here, the loss function is computed as the

following cross-entropy loss.

$$L_{LM} = -\sum_{n=1}^{|y|} \log P_{\mathcal{G}}(y_n \,|\, y_{<n},\, v,\, q_m). \tag{9}$$

Here, y_n is the n-th word of the ground-truth sub-question.

To further improve the informativeness of the generated sub-questions, we train the VQG model with a reinforcement learning (RL) loss, that uses the info-score obtained by the info-score model as a reward. The reinforcement learning loss is expressed as the following equation.

$$L_{RL} = -(\hat{I} - \hat{I}_b) \sum \log P_{\mathcal{G}}(y_n \,|\, y_{<n},\, v,\, q_m) \tag{10}$$

We follow self-critical sequence training method [28] for the calculation of L_{RL}. Here, $\hat{I}_b$ is the baseline reward that is introduced to stabilize the training, which is, the result of the info-score for the greedy-decoded sub-question. $\hat{I}$ is the info-score computed for the questions generated by sampling based on the multinomial distribution of the words output by the model. Thus, the model is motivated to generate sub-questions that have a higher info-score than those generated by greedy-decoding.

Training with RL loss alone may increase the info-score; however, this loss does not consider the fluency of the output sentences, which may corrupt the generated sub-questions. To avoid this, we train the model with a combination of cross-entropy loss and RL loss, that is,

$$L = \gamma L_{LM} + (1 - \gamma) L_{RL} \tag{11}$$

where γ is the hyperparameter for balancing the loss values.

3.4. Teacher VQA Model

Ideally, our pipeline should be run in a human-in-the-loop setting, where humans provide answers to questions generated by the VQG model. However, it is not feasible to ask humans to answer all of the generated questions each time we conduct a model evaluation. Therefore, we create a Teacher VQA model $\mathcal{A}_T$ that provides sub-answers to be fed to the Target VQA model.

The architecture of the Teacher VQA model is essentially the same as that of the Target VQA model. The Teacher VQA model must have fairly high performance due to its role of providing sub-answers on behalf of humans. Therefore, we feed the model with the ground-truth main-answer as additional oracle information, i.e., $a_s = \mathcal{A}_T(v, [q_s; q_m; a_m])$. Note that the oracle main-answer is only available when the Teacher VQA model provides sub-answers and is not seen from the Target VQA model. This model is trained to minimize the binary cross entropy loss as follows:

$$\begin{aligned} L_{\mathcal{T}} = &- a_s \log P_{\mathcal{A}_{\mathcal{T}}}(a_s \,|\, v,\, q_m,\, a_m,\, q_s) \\ &- (1 - a_s) \log(1 - P_{\mathcal{A}_{\mathcal{T}}}(a_s \,|\, v,\, q_m,\, a_m,\, q_s)) \end{aligned} \tag{12}$$

	yes/no	what	other	all
train	161,140	18,447	19,567	199,154
val	17,894	2,055	2,191	22,140

Table 1. Number of questions per type after re-splitting the VQA-Introspect dataset.

4. Experiment

4.1. Dataset

We used VQA-Introspect [29] dataset for our experiments. The VQA-Introspect is a dataset constructed by additionally annotating the perceptual sub-questions for reasoning questions included in the VQA v1 and v2 datasets [3, 11]. We associated the main-question and sub-question in their dataset in a one-to-one manner, and obtained 199,154 (main-question, sub-question) pairs as training data and 22,140 pairs as validation data. For detailed analysis of the results, we categorized the main-question types into "yes/no", "what", and "other". The number of questions per type are listed in Table. 1.

4.2. Implementation Details

The encoder for all models is the Transformer encoder, which has a common structure based on UNITER. We use the bottom-up top-down image features [4] extracted by Faster R-CNN [27]. The number of object regions is adaptively set between 10 and 100, and the feature dimension is set to 2,048. The number of Transformer blocks in the encoder and decoder is set to 12, and the number of hidden units in the each Transformer block is set to 768. We use the AdamW optimizer [21] with the parameters ($\beta_1 = 0.9$, $\beta_2 = 0.999$). The initial learning rate is set to 8.0×10^{-5} for the Target VQA model, 3.0×10^{-5} for the Info-score model, and 5.0×10^{-5} for the VQG model.

We employ the cosine annealing scheduling, where warm-up is performed for 10% of the entire training steps. We train the Target VQA model for 12K steps with a batch size of 84, Info-score model for 3K steps with a batch size of 48. For the VQG model, we train the model with only L_{LM} for the first 30 epochs, and then train it also with L_{RL} for the following six epochs. For the hyper-parameter γ, which balances the LM loss and RL losses, we run experiments with $\gamma \in \{0.05, 0.1, 0.5\}$ and report the results for the case $\gamma = 0.5$, which gives the best results. The training of the VQG model took approximately 17 hours on a single Tesla A100 GPU.

	Accuracy				*Comparison with UNITER baseline*	
	Overall	yes / no	what	others	UNITER ✗ \ Ours ✓ (↑)	UNITER ✓ \ Ours ✗ (↓)
UNITER (baseline)	69.71	76.97	45.73	46.08	-	-
Ours	**77.12**	**85.60**	47.22	50.77	**10.75**	**3.338**
Ours w/o RL	76.20	84.20	**47.61**	**51.54**	10.07	3.582
Ours w/o answer	70.21	77.33	46.02	47.46	4.313	3.812
Ours with GT	81.92	87.46	59.34	66.76	14.25	2.03

Table 2. Evaluation based on the performance of the Target VQA model when it is given the questions generated by each model. The left columns show the accuracy of the overall evaluation set and the accuracy of each question type. The right columns show the ratio of the Target VQA model to answer correct/ wrong answers when compared to the UNITER baseline (i.e., without sub-questions). In the former case (UNITER ✗ \ +sub ✓), a larger number indicates better performance; in the latter case (UNITER ✓ \ +sub ✗), a smaller number indicates better performance.

4.3. Training Details of Info-score Model

We calculate the ground-truth info-score I_{GT} from a pre-trained Target VQA model to train the Info-score model. It should be noted that the dataset used to train the Target VQA model contained sub-questions that were considered to be related to the main-question to some extent. This means that the training dataset is unlikely to contain any sub-questions that are useless for the main-question, which may result in a significant data imbalance in the info-score.

To avoid this imbalance problem, we perform mini-batch negative sampling. During training, for a given main-question, we randomly select a sub-question associated with a different main-question in the same mini-batch to be the negative sample sub-question. When a positive sample is input to the model, the model is trained to minimize the binary cross-entropy loss calculated on the ground-truth distribution of the main-answer, as in any other standard VQA model. When a negative example sub question is input to the model, we train the model to output a uniform distribution with all zero values.

4.4. Compared Approaches

We compare our model with **UNITER** baseline, where the Target VQA model makes predictions from the main-question only. In this setup, no additional information is input other than the main-question, which is the same as the usual experimental setup for VQA that has been used in many existing studies.

In addition, we conduct ablation studies with following ablation models: **w/o answer** and **w/o RL**. In **w/o answer** setting, we do not use the sub-answer predicted by Teacher VQA model. This enables us to investigate whether the use of sub-answers as well as sub-questions is intrinsically important in terms of acquiring information about the main-question. Specifically, we use a special unknown token, instead of the answer from the Teacher VQA model, as an input to the Target VQA model. In **w/o RL** setting, the Target VQA model uses the sub-question generated by the VQG model as an additional input, but in this case, the VQG model is not trained on the RL loss, but only on the LM loss. This experimental setup was designed to see whether RL loss with info-score as a reward would generate more informative question.

Finally, we report the result of **with GT** setting. In this setting, the Target VQA model uses the ground-truth sub-question as an additional input. We can consider this to be an upper bound, since sub-questions and sub-answers are provided by humans, not by the model.

4.5. Results and Discussions

4.5.1 Performance of the Target VQA Model

Table 2 lists the performance of the model compared to other approaches.

First, we evaluate how the performance of the Target VQA model changes when the sub-questions generated by the VQG model are used as additional inputs. The results show that using the sub-questions generated by the VQG model as additional information indeed helps to improve the performance of the Target VQA model (see UNITER vs. the others). Further, adding RL loss, which uses the info-score as an reward, can improve the effectiveness of the sub-question (see Ours vs. Ours w/o RL). The results further show that, it is important to have appropriate answers to the sub-questions to obtain information (see Ours vs. Ours w/o answer). Also, in the case of Ours w/o answer, there is some improvement in performance compared to UNITER baseline. This is probably because the generated sub-questions themselves also contain additional information about the main-questions.

We further show the accuracy for each question type. The methods that add sub-questions (Ours w/o RL, Ours) show significant performance improvement compared to UNITER baseline for "yes/no" type. For "what" and "other" type questions, our method with RL shows slight

	BLEU-4	Info-score (↑)	*Comparison with UNITER baseline per sub-question type* yes/no Num.	yes/no ΔAcc.(%)	what Num.	what ΔAcc.(%)	other Num.	other ΔAcc.(%)
w/o RL	18.25	1.88×10^{-5}	19,868	10.08	2,110	0.25	162	**8.64**
Ours	**18.66**	$\mathbf{3.58 \times 10^{-5}}$	20,529	**11.09**	1,457	**3.33**	154	5.68
Ours with GT	-	1.03×10^{-4}	19,553	17.82	2,008	13.53	579	21.08

Table 3. Evaluation of the VQG model in terms of the quality of the generated questions, the informativeness, and the properties of the generated sub-questions. In the right columns, the number of sub-questions generated is shown for each question type. In addition, for each question type, the column ΔAcc. shows an improvement in the accuracy of the main-answer predictions compared to the baseline.

performance drop compared to UNITER baseline and w/o RL. The importance of sub-answer can be seen from the fact that when there is no appropriate answer to the sub-question, the performance degrades for all question types.

We also analyze how adding sub-questions changes the results compared to the UNITER baseline. We observe the same trend here, as in the previous experiments, that models using reinforcement learning outperform the others. In the case where the model makes incorrect predictions in UNITER baseline and correct predictions with sub-questions (UNITER ✗ \ +sub ✓), the results are better in Ours with RL than w/o RL and w/o answer. When the model answered correctly in UNITER baseline and incorrectly with sub-question (UNITER ✓ \ +sub ✗), the VQG model with RL also performed better than the model without RL and without answer.

4.5.2 Performance of the VQG Model

In this section, we discuss the performance of the VQG model. The results are listed in Table 3.

When compared with the BLEU score, which indicates how well the correct and generated sentences match, the model w/o RL has a slightly higher value. As for the info-score, which indicates how informative the generated sub-questions are, we can see that our model with RL can indeed obtain a higher info-score. From these results, we can say that our model is successful in generating more informative questions without sacrificing the fluency of the generated sentences. The right columns of the table list the properties of the generated sub-questions categorized by question type. The values of ΔAcc. indicates an improvement in the accuracy of the predictions by the Target VQA model compared to the UNITER baseline. Based on the value of ΔAcc. of GT, the question types of sub-questions that help improve the performance of the Target VQA model are other > yes/no > what. Comparing the number of generated sub-questions between Ours w/o RL and Ours, the number of "yes/no" type questions increased in Ours, while the number of "what" type questions decreased. From these results, it can be said that our model tends to generate more questions that are likely to improve the performance of the Target VQA model, and fewer questions that are not likely to contribute to performance improvement. As for the "other" type questions, there are few questions even in GT, thus, the VQG model probably could not learn to generate them in large numbers.

4.6. Qualitative Results

In Figure 3, we show some examples of the generated sub-questions, the sub-answer provided by the Teacher VQA model, and the final prediction made by the Target VQA model. Generally, we can say that our VQG model is able to generate perceptual sub-questions related to the image and the main-question. In some cases, the model without RL generates questions that are related to the image and the main-question, but do not help to answer the main-question (e.g., "is there a fork on the plate?" in the upper left example). Even in such a case, we can see that our model is able to generate informative questions (e.g., "is the fork on the left?"). In other cases, as in the example in bottom right, the VQG model may generate a question that is relevant to the image but not much relevant to the main-question. One possible reason for this is that the multi-modal encoder of the VQG model may not be able to encode the text information properly. We believe that we can further improve the sub-question generation that reflects the content of the main-question by developing an encoder that better encodes the image and text information.

5. Conclusion

In this study, we propose an informative perceptual sub-question generation method to improve the VQA performance. We define the info-score to quantitatively measure the usefulness of sub-questions, and created a model to estimate the info-score from the main-questions and sub-questions. By training our model using reinforcement learning with info-score as a reward, we were able to acquire useful information to answer the main-question and achieve performance improvement of the VQA model.

Future challenges include to extend this research to ac-

main-question : on which side of the plate is the fork?
sub-question : is there a tined utensil to the left of the pizza?

GT main-answer : left
UNITER baseline pred : ✗ right

===== w/o RL =====
sub-question : is there a fork on the plate?
sub-answer (by oracle) : yes
main-answer : ✗ right

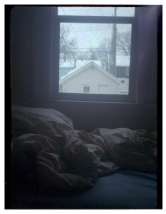
===== Ours =====
sub-question : is the fork on the left?
sub-answer (by oracle) : yes
main-answer : ✓ left

main-question : how bright is the room?
sub-question : is there light reflecting off the door?

GT main-answer : very
UNITER baseline pred : ✗ bright

===== w/o RL =====
sub-question : are there any lights on in the room?
sub-answer (by oracle) : yes
main-answer : ✗ bright

===== Ours =====
sub-question : is there a lot of light in the room?
sub-answer (by oracle) : yes
main-answer : ✓ very



main-question : what season of the year is it?
sub-question : is there snow?

GT main-answer : winter
UNITER baseline pred : ✗ summer

===== w/o RL =====
sub-question : is there snow on the ground?
sub-answer (by oracle) : yes
main-answer : ✓ winter

===== Ours =====
sub-question : is there snow outside?
sub-answer (by oracle) : yes
main-answer : ✓ winter

main-question : which seems the most modern element in the room?
sub-question : what object in the room is electric?

GT main-answer : microwave
UNITER baseline pred : ✗ on right

===== w/o RL =====
sub-question : is there a microwave on the counter?
sub-answer (by oracle) : yes
main-answer : ✓ microwave

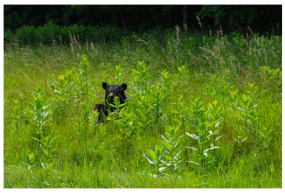
===== Ours =====
sub-question : is there a microwave in the kitchen?
sub-answer (by oracle) : yes
main-answer : ✓ microwave

main-question : how well are the elephants able to hide?
sub-question : is the surrounding vegetation taller than the elephants?

GT main-answer : not well
UNITER baseline pred : ✗ very

===== w/o RL =====
sub-question : are the elephants in the wild?
sub-answer (by oracle) : yes
main-answer : ✗ crossed

===== Ours =====
sub-question : is there a lot of space between the elephants?
sub-answer (by oracle) : no
main-answer : ✗ crossed



main-question : what is peeking out of the grass?
sub-question : what color is the bear?

GT main-answer : bear
UNITER baseline pred : ✗ cow

===== w/o RL =====
sub-question : what color is the grass?
sub-answer (by oracle) : green
main-answer : ✗ cow

===== Ours =====
sub-question : is there grass on the ground?
sub-answer (by oracle) : yes
main-answer : ✗ cow

Figure 3. Some qualitative examples of the results. The top part of each result shows the ground-truth annotations and the prediction result by the Target VQA model in UNITER baseline. The middle part shows the sub-question generated by our model, the sub-answer predicted by the Teacher VQA model, and the prediction results of the Target VQA model with additional inputs of the sub-question and the sub-answer. In the lower part, the results using the sub-questions generated by the model without RL loss are shown.

quire information by generating multiple sub-questions. However, asking too many questions would be burdensome to the answerer, thus, future research will require to consider the trade-off between efficiency (i.e., how few questions to ask) and accuracy.

Acknowledgements

This work was supported by D-CORE Grant from Microsoft Research Asia and partially supported by JST AIP Acceleration Research JPMJCR20U3, Moonshot R&D Grant Number JPMJPS2011, JSPS KAKENHI Grant Number JP19H01115, and JP20H05556 and Basic Research Grant (Super AI) of Institute for AI and Beyond of the University of Tokyo. We would like to thank Naoyuki Gunji and Hiroaki Yamane for the helpful discussions.

References

[1] Vedika Agarwal, Rakshith Shetty, and Mario Fritz. Towards causal vqa: Revealing and reducing spurious correlations by invariant and covariant semantic editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 2

[2] Aishwarya Agrawal, Dhruv Batra, Devi Parikh, and Aniruddha Kembhavi. Don't just assume; look and answer: Overcoming priors for visual question answering. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4971–4980, 2018. 2

[3] Aishwarya Agrawal, Jiasen Lu, Stanislaw Antol, Margaret Mitchell, C. Lawrence Zitnick, Devi Parikh, and Dhruv Batra. Vqa: Visual question answering. *International Journal of Computer Vision*, 123:4–31, 2015. 1, 5

[4] Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. Bottom-up and top-down attention for image captioning and visual question answering. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6077–6086, 2018. 5

[5] Pratyay Banerjee, Tejas Gokhale, Yezhou Yang, and Chitta Baral. WeaQA: Weak supervision via captions for visual question answering. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 3420–3435, 2021. 2

[6] Rémi Cadène, Hedi Ben-younes, Matthieu Cord, and Nicolas Thome. Murel: Multimodal relational reasoning for visual question answering. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1989–1998, 2019. 2

[7] Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. UNITER: Universal image-text representation learning. In *European conference on computer vision (ECCV)*, pages 104–120. Springer, 2020. 1, 4

[8] Cui cui Zhang, Wei-Lun Chao, and Dong Xuan. An empirical study on leveraging scene graphs for visual question answering. In *British Machine Vision Conference (BMVC)*, 2019. 2

[9] Tejas Gokhale, Pratyay Banerjee, Chitta Baral, and Yezhou Yang. MUTANT: A training paradigm for out-of-distribution generalization in visual question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 878–892, Online, Nov. 2020. Association for Computational Linguistics. 2

[10] Tejas Gokhale, Pratyay Banerjee, Chitta Baral, and Yezhou Yang. Vqa-lol: Visual question answering under the lens of logic. In *European conference on computer vision*, pages 379–396. Springer, 2020. 2

[11] Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6325–6334, 2017. 1, 5

[12] Yu Jiang, Vivek Natarajan, Xinlei Chen, Marcus Rohrbach, Dhruv Batra, and Devi Parikh. Pythia v0.1: the winning entry to the vqa challenge 2018. *ArXiv*, abs/1807.09956, 2018. 1

[13] Kushal Kafle, Mohammed Yousefhussien, and Christopher Kanan. Data augmentation for visual question answering. In *Proceedings of the 10th International Conference on Natural Language Generation*, pages 198–202, Sept. 2017. 2

[14] Yash Kant, Abhinav Moudgil, Dhruv Batra, Devi Parikh, and Harsh Agrawal. Contrast and classify: Training robust vqa models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1604–1613, October 2021. 2

[15] Hyounghun Kim and Mohit Bansal. Improving visual question answering by referring to generated paragraph captions. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3606–3612, July 2019. 2

[16] Wonjae Kim, Bokyung Son, and Ildoo Kim. Vilt: Vision-and-language transformer without convolution or region supervision. In *ICML*, 2021. 4

[17] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, July 2020. 4

[18] Hui Li, Peng Wang, Chunhua Shen, and Anton van den Hengel. Visual question answering as reading comprehension. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6312–6321, 2019. 2

[19] Qing Li, Jianlong Fu, Dongfei Yu, Tao Mei, and Jiebo Luo. Tell-and-answer: Towards explainable visual question answering using attributes and captions. In *EMNLP*, 2018. 2

[20] Xiujun Li, Xi Yin, Chunyuan Li, Xiaowei Hu, Pengchuan Zhang, Lei Zhang, Lijuan Wang, Houdong Hu, Li Dong, Furu Wei, Yejin Choi, and Jianfeng Gao. Oscar: Object-semantics aligned pre-training for vision-language tasks. In *ECCV*, 2020. 4

[21] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019. 5

[22] Aroma Mahendru, Viraj Prabhu, Akrit Mohapatra, Dhruv Batra, and Stefan Lee. The promise of premise: Harnessing question premises in visual question answering. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 926–935, Copenhagen, Denmark, Sept. 2017. Association for Computational Linguistics. 1

[23] Kenneth Marino, Xinlei Chen, Devi Parikh, Abhinav Gupta, and Marcus Rohrbach. Krisp: Integrating implicit and symbolic knowledge for open-domain knowledge-based vqa. In *CVPR*, 2021. 2

[24] Ishan Misra, Ross B. Girshick, Rob Fergus, Martial Hebert, Abhinav Gupta, and Laurens van der Maaten. Learning by asking questions. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11–20, 2018. 2

[25] Medhini Narasimhan, Svetlana Lazebnik, and Alexander G. Schwing. Out of the box: Reasoning with graph convolution

nets for factual visual question answering. In *NeurIPS*, 2018. 2

[26] Medhini Narasimhan and Alexander G Schwing. Straight to the facts: Learning knowledge base retrieval for factual visual question answering. In *Proceedings of the European conference on computer vision (ECCV)*, pages 451–468, 2018. 2

[27] Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39:1137–1149, 2015. 5

[28] Steven J. Rennie, Etienne Marcheret, Youssef Mroueh, Jerret Ross, and Vaibhava Goel. Self-critical sequence training for image captioning. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1179–1195, 2017. 5

[29] Ramprasaath R. Selvaraju, Purva Tendulkar, Devi Parikh, Eric Horvitz, Marco Túlio Ribeiro, Besmira Nushi, and Ece Kamar. Squinting at vqa models: Introspecting vqa models with sub-questions. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10000–10008, 2020. 1, 2, 5

[30] Meet Shah, Xinlei Chen, Marcus Rohrbach, and Devi Parikh. Cycle-consistency for robust visual question answering. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6642–6651, 2019. 2

[31] Tingke Shen, Amlan Kar, and Sanja Fidler. Learning to caption images through a lifetime by asking questions. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 10392–10401, 2019. 2

[32] Jiaxin Shi, Hanwang Zhang, and Juan-Zi Li. Explainable and explicit visual reasoning over scene graphs. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8368–8376, 2019. 2

[33] Alane Suhr, Stephanie Zhou, Ally Zhang, Iris Zhang, Huajun Bai, and Yoav Artzi. A corpus for reasoning about natural language grounded in photographs. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6418–6428, July 2019. 4

[34] Ruixue Tang, Chao Ma, Wei Emma Zhang, Qi Wu, and Xiaokang Yang. Semantic equivalent adversarial data augmentation for visual question answering. In *European Conference on Computer Vision*, pages 437–453, 2020. 2

[35] Damien Teney, Peter Anderson, Xiaodong He, and Anton van den Hengel. Tips and tricks for visual question answering: Learnings from the 2017 challenge. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4223–4232, 2018. 1, 4

[36] Zixu Wang, Yishu Miao, and Lucia Specia. Cross-modal generative augmentation for visual question answering. *arXiv preprint arXiv:2105.04780*, 2021. 2

[37] Spencer Whitehead, Hui Wu, Yi Ren Fung, Heng Ji, Rogério Schmidt Feris, and Kate Saenko. Learning from lexical perturbations for consistent visual question answering. *ArXiv*, abs/2011.13406, 2020. 2

[38] Jialin Wu, Zeyuan Hu, and Raymond J. Mooney. Generating question relevant captions to aid visual question answering. *ArXiv*, abs/1906.00513, 2019. 3

[39] Jialin Wu and Raymond Mooney. Self-critical reasoning for robust visual question answering. *Advances in Neural Information Processing Systems*, 32, 2019. 2

[40] Pengchuan Zhang, Xiujun Li, Xiaowei Hu, Jianwei Yang, Lei Zhang, Lijuan Wang, Yejin Choi, and Jianfeng Gao. Vinvl: Revisiting visual representations in vision-language models. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5575–5584, 2021. 4