Improving Zero-shot Visual Question Answering via Large Language Models with Reasoning Question Prompts

Yunshi Lan
East China Normal University
Shanghai, China
yslan@dase.ecnu.edu.cn

Xiang Li
East China Normal University
Shanghai, China
xiang.li@stu.ecnu.edu.cn

Xin Liu
East China Normal University
Shanghai, China
xinliu@stu.ecnu.edu.cn

Yang Li
Alibaba Group
Beijing, China
ly200170@alibaba-inc.com

Wei Qin
Hefei University of Technology
Hefei, China
qinwei.hfut@gmail.com

Weining Qian
East China Normal University
Shanghai, China
wnqian@dase.ecnu.edu.cn

ABSTRACT

Zero-shot Visual Question Answering (VQA) is a prominent vision-language task that examines both the visual and textual understanding capability of systems in the absence of training data. Recently, by converting the images into captions, information across multi-modalities is bridged and Large Language Models (LLMs) can apply their strong zero-shot generalization capability to unseen questions. To design ideal prompts for solving VQA via LLMs, several studies have explored different strategies to select or generate question-answer pairs as the exemplar prompts, which guide LLMs to answer the current questions effectively. However, they totally ignore the role of question prompts. The original questions in VQA tasks usually encounter ellipses and ambiguity which require intermediate reasoning. To this end, we present Reasoning Question Prompts for VQA tasks, which can further activate the potential of LLMs in zero-shot scenarios. Specifically, for each question, we first generate self-contained questions as reasoning question prompts via an unsupervised question edition module considering sentence fluency, semantic integrity and syntactic invariance. Each reasoning question prompt clearly indicates the intent of the original question. This results in a set of candidate answers. Then, the candidate answers associated with their confidence scores acting as answer heuristics are fed into LLMs and produce the final answer. We evaluate reasoning question prompts on three VQA challenges, experimental results demonstrate that they can significantly improve the results of LLMs on zero-shot setting and outperform existing state-of-the-art zero-shot methods on three out of four data sets. Our source code is publicly released at https://github.com/ECNU-DASE-NLP/RQP.

CCS CONCEPTS

• **Computing methodologies** → **Artificial intelligence**; • **Information systems** → **Multimedia and multimedial retrieval**.

MM '23, October 29-November 3, 2023, Ottawa, ON, Canada

ACM ISBN 979-8-4007-0108-5/23/10.
https://doi.org/10.1145/3581783.3612389

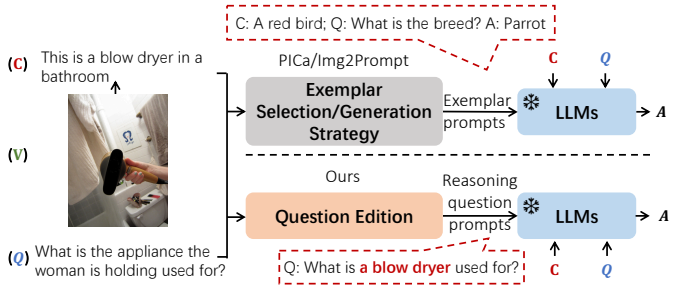


Figure 1: Comparison between existing prompting methods and our method on VQA tasks using frozen LLMs [6, 56]. The images are first converted into captions. Prior studies proposed different strategies to select exemplars from training data like PICa [54] or generate synthetic exemplars like Img2Prompt [16]. In contrast, our method focuses on question prompt generation, where self-contained questions are produced in an unsupervised manner such that LLMs can easily capture the intent of questions and fully exert their potential.

KEYWORDS

visual question answering, zero-shot evaluation, large language models

ACM Reference Format:
Yunshi Lan, Xiang Li, Xin Liu, Yang Li, Wei Qin, and Weining Qian. 2023. Improving Zero-shot Visual Question Answering via Large Language Models with Reasoning Question Prompts. In *Proceedings of the 31st ACM International Conference on Multimedia (MM '23), October 29-November 3, 2023, Ottawa, ON, Canada.* ACM, New York, NY, USA, 13 pages. https://doi.org/10.1145/3581783.3612389

1 INTRODUCTION

Visual Question Answering (VQA) tasks require a system to answer a textual question about an image. Diverse studies focused on solving visual questions, the answer of which can be directly derived from the image[3], or questions requiring outside knowledge beyond the image content [32, 49]. Due to the enormous demand

of manpower to annotate VQA datasets and the risk of human biases [1, 7], there are quite a few studies proposing methods to solve zero-shot VQA tasks, where no image-question pair is provided for training [4, 17, 44].

To solve zero-shot VQA tasks, early studies developed methods to synthesize training data so that conventional VQA models can be trained on the synthetic data [7, 8, 17, 44]. Recently, Large Language Models (LLMs), which are trained on general text corpus, have shown excellent generalization capability on zero-shot tasks, such as information extraction [51] and logical reasoning [59]. Inspired by the intriguing properties of LLMs, Yang et al. (2022) first proposed PICa, which transfers images into captions then a frozen off-the-shelf LLM is applied to answer the question based on the caption context. It not only saves the effort of pre-training a multi-modal model, but also provides world-knowledge to answer the questions. Take the question in Figure 1 as an example, the image is converted into the caption "*This is a blow dryer in a bathroom.*". The question "*What is the appliance the woman is holding used for?*" should be answered with the caption as context. To guide LLMs to better understand the tasks, in-context examples are selected from the training data as the prompts. Soon after, another study proposes Img2Prompt [16], which generates synthetic question-answer pairs via template-based and neural question-generation methods based on the images, which has shown impressive performance on zero-shot VQA benchmark datasets.

We observe that most of the existing prompting methods on VQA tasks focus on developing different exemplar selection/generation strategies to help LLMs better comprehend the task thus enhancing its capacity. However, there is a demand of eliminating the semantic gap between captions and questions, which can be illustrated from two aspects: (1) The current methods entirely rely on the understanding capability of LLMs to resolve the ambiguity and infer the intent of the questions, which might involve unexpected bias [21, 41]. As we can see, in Figure 1, the question asks about "*this appliance*", which indicates "*blow dryer*". Due to the bias existing in LLMs, they may fail to parse the question correctly. (2) LLMs are brittle to ill-posed questions, especially under the zero-shot setting. In Figure 1, "*the woman is holding*" is irrelevant to the image. LLMs are sensitive to such noisy information and it may cause confusion to LLMs [58]. In this case, disambiguating the question is of high demand.

Motivated by the observation, we present **RQ** prompts, which are **R**easoning **Q**uestion prompts for improving the understanding capability of LLMs under zero-shot VQA scenarios. Specifically, we design an unsupervised question edition module to convert original questions into self-contained questions by editing the segments of the question. We propose a search algorithm to generate the possible edited questions and rank them by a scoring function. The scoring function measures sentence fluency, semantic integrity and syntactic invariance. Eventually, the top-ranked reasoning question prompts are utilized to generate a set of candidate answers. Following the heuristic prompting in Prophet [43], where prompting is divided into answer generation and answer choosing steps, we encode both answer candidates and a confidence score to form answer heuristics for choosing. The confidence score takes both the confidence of the reasoning question prompt and the generated answer into consideration, which produces a comprehensive score for choosing. Our contribution can be summarized as follows:

- We propose RQ prompts, which aim to improve zero-shot VQA tasks via LLMs by providing edited questions as prompts. No extra data as well as supervision is needed for the RQ prompts generation procedure.
- We design a novel confidence scoring function for the answer heuristics, which can comprehensively measure the answer candidates.
- Reasoning question prompts generally improve existing baselines with absolute improvement ranging from 0.3 to 5.2 points. Our method achieves new state-of-the-art results on three out of four evaluated zero-shot VQA data sets.

2 RELATED WORK

2.1 VQA tasks

Given a textual question, VQA tasks require a system to answer the question by decoding the information from an image and even utilizing external knowledge. Several benchmark datasets [32, 42, 48, 49], including complex reasoning questions, facilitate the development of this field. To incorporate with external knowledge, early methods turned to textual Knowledge Bases (KBs) and applied either graph-based [24, 36, 60, 61] or transformer-based approaches [11, 13] to introduce the KB information into the question answering module. Besides, multi-modal KBs are also leveraged to solve VQA tasks. Wu et al. (2022) combine Wikipedia, ConceptNet and Google images to supplement multi-modal knowledge. With the emergence of language models, researchers consider them as implicit KBs [43, 54] and there are several studies [12, 15, 28, 31] combining explicit and implicit knowledge to improve model's ability of handling visual questions. Recently, large language models impress people by their quantum leap of understanding and reasoning capabilities. Several studies [43, 54] reformulate VQA tasks into a textual question answering task by converting the images into captions and apply in-context learning to activate the implicit knowledge in LLMs [6]. In this paper, we discuss VQA tasks under zero-shot scenarios, which brings in new challenges to the tasks.

2.2 Zero/Few shot of VQA tasks

There is a line of work focusing on solving zero/few-shot VQA tasks. A general solution is to augment image-question pairs for training. Multi-modal pre-training models like CLIP [38] are frequently leveraged to generate synthetic question-answer pairs from images [4, 7]. After that, a VQA model can be trained with the augmented data so that it can learn patterns and answer questions in the test set. Tsimpoukelli et al. (2021b) simply train a vision encoder to represent each image as a sequence of continuous embeddings, which could collaborate well with a frozen language model. This inspires more studies [2, 17, 26, 30] proposing parameter-efficient methods to combine both pre-trained vision models and language models for zero/few-shot VQA. Guo et al. (2023) shift to the paradigm of leveraging LLMs to solve VQA tasks, they propose a method to automatically generate prompts as exemplars under the zero-shot setting. This is the closest study to our work, but our work is different as we focus on the question prompts instead of exemplar prompts.

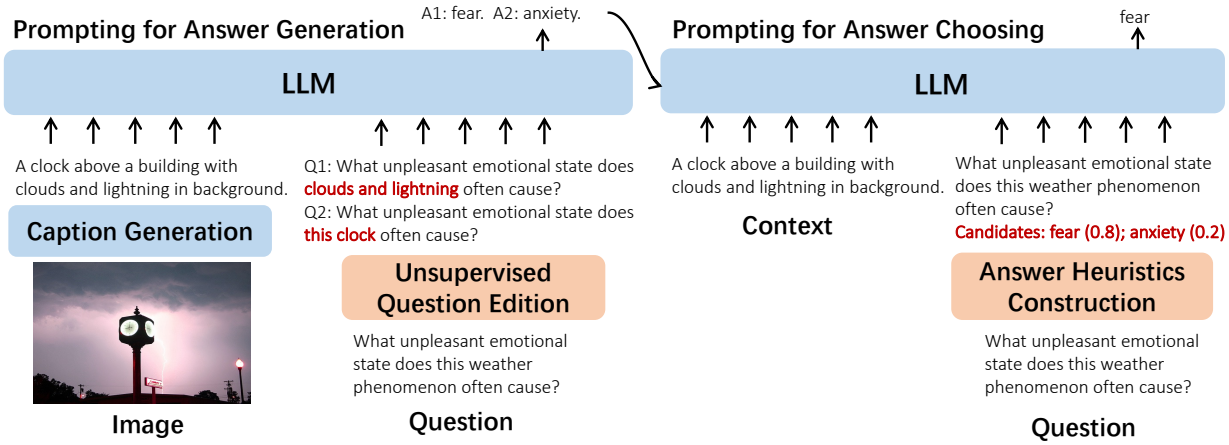


Figure 2: The illustration of our prompting method that enables LLMs to perform VQA tasks with two-step reasoning. The blue blocks denote the modules with frozen parameters and the orange blocks denote the modules we propose to generate reasoning question prompts and answer heuristics.

2.3 Prompt Tuning of LLMs

Prompts are significant regarding the inference of LLMs. It helps guide the LLMs to activate the potentials of understanding and reasoning. A question could be part of the prompt. It should be well designed to fit the nature of the evaluated tasks [40]. For example, pattern-verbalizer pair is one type of question prompt which maps diverse tasks into a word prediction task. Besides, there are some other prompts. An instructional prompt primarily contains a natural language description of the underlying task. Generally, a narrative sentence is annotated manually as the instruction prompt [35, 50]. Recently, researchers decompose a complex task into sub-tasks so that the multiple instruction prompts guide LLMs to handle sub-tasks step by step [22, 49, 52, 57, 59]. An exemplar prompt guides LLMs by showing some examples from the training data. There are a number of studies proposing different strategies to select or generate good exemplar prompts for LLMs [6, 20, 29, 34]. Instead of discrete text, prompts could be in the format of continuous embeddings, researchers have developed diverse methods to learn better embeddings [25, 37]. Our work takes effort on improving the question prompt by eliminating the semantic gap between the original question and images for zero-shot VQA tasks.

3 METHODS

3.1 Overview

In this section, we introduce our prompting method for solving zero-shot VQA tasks. Following Prophet [43], which is a heuristic prompting framework, we also decompose the task into two steps as shown in Figure 2. In prompting step for answer generation, we convert an image into a caption with a frozen caption model [55] as the context of the given question. Particularly, we edit the question with an unsupervised method, namely **Unsupervised Question Edition** module, to transfer the original question into the reasoning question prompts. For each reasoning question prompt, we generate a candidate answer from a frozen LLM. In prompting step for answer choosing, we construct answer heuristics via **Answer Heuristics Construction** module based on the candidate answers generated above. Then a frozen LLM is required to choose correct answer among these candidates. Each candidate answer in the prompt is associated with a confidence score taking account of the confidence of both question prompts and answers.

3.2 Prompting for Answer Generation

To bridge the gap between the image captions and questions, we generate reasoning question prompts to avoid errors resulting from missing reasoning step. Then we generate candidate answers based on them. We define the reasoning question prompts should meet the following criterias:

- The generated questions should not contain any ellipsis and ambiguity. In other words, they should be self-contained. Such that LLMs could easily understand the question without guessing the implicit information. Take the question in Figure 2 as an example, "*this weather phenomenon*" in question should be explicated by "*clouds and lightning*".
- The self-contained question should be produced in the absence of supervision signal under a zero-shot setting. A neural network-based model is difficult to be applied as it requires a large volume of labeled data to learn how to generate a self-contained question.

To meet the above criterias, we propose an unsupervised method to edit the original question with the consideration of its image caption. There are two advantages of conducting edition on the original questions instead of generation: (1) It is controllable to revise the original questions by substitution. Only segments of the questions can be changed and the major semantics of the original questions is maintained. (2) Even without parallel labeled data, it is possible to conduct edition on the original question by a search algorithm holding a search objective. On this basis, we design an unsupervised question edition module to convert the original question into a reasoning question prompt.

Unsupervised Question Edition. Inspired from existing work on text simplification [23], we design an edit-based search algorithm to produce the reasoning question prompts by conducting substitution

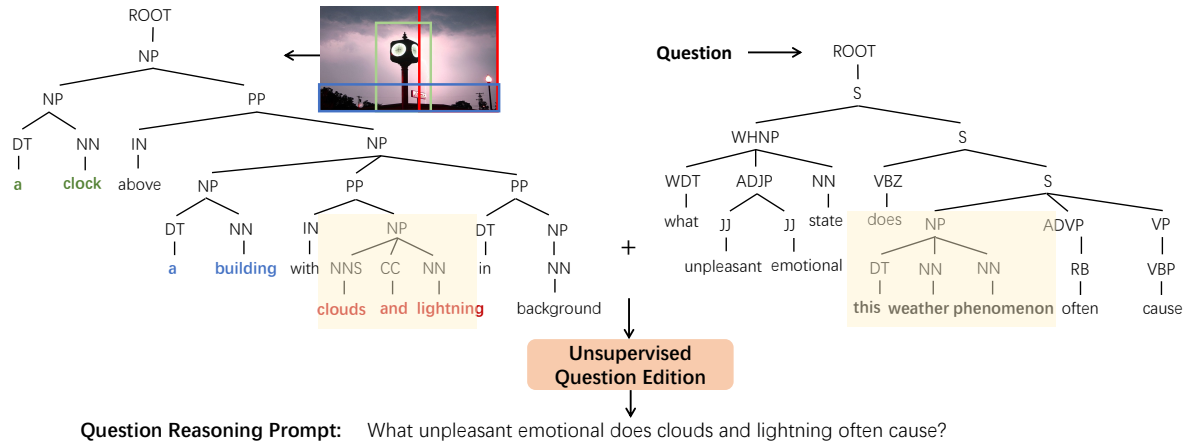


Figure 3: The generation process of reasoning question prompts in unsupervised question edition module. Both the question and caption are transformed into constituency parse trees. The phrase-level constituents in the caption correspond to the different objects in the image, which are shown with different colors. They would be utilized to substituent segments of the original question to form a complete self-contained question. The yellow shades indicate that we substituent these constituents to form a reasoning question prompt.

operations on the constituency parse tree. As shown in Figure 3, "*clouds and lightning*" (NP) and "*this weather phenomeon*" (NP) are both considered as phrase-level constituents with the same root tag "*Noun Phrase*" based on the constituency parse trees. By replacing "*this weather phenomenon*" with "*clouds and lightning*", we can obtain a self-contained question "*what unpleasant emotional does clouds and lightning often cause?*".

Given a caption and a question, our search algorithm iteratively performs edits to search for a candidate. Specifically, starting from the constituents of the caption, we consider all the constituents of the original question and conduct substitution to generate candidates. Each candidate will be measured by a scoring function considering the sentence fluency, semantics integrity and syntactic invariance. The candidate with the score higher than a threshold can be saved and further edited. The detailed search algorithm is displayed in Algorithm 1.

Algorithm 1 Search Algorithm of Reasoning Question Prompts

procedure QE(C, Q) ▷ C and Q are the parse trees of captions and original questions, respectively.
  $\mathcal{S} = \{Q\}, \mathcal{S}_{batch} = \{Q\}$
  for $j = 1, ..., l$ **do** ▷ l: number of constituents in C
    $\mathcal{S}_{best} = \{\}$ ▷ Initialization
    for Q' in $\mathcal{S}_{batch}$ **do**
      for $i = 1, ..., n$ **do** ▷ n: number of constituents in Q
        $\tilde{Q} \leftarrow substituent(Q'[i], C[j])$ ▷ Edit constituents
        $s \leftarrow f(\tilde{Q})$ ▷ Score the above candidate
        if $s > (f(Q) - \rho)$ **then**
          $\mathcal{S}_{best} \leftarrow \mathcal{S}_{best} \cup \{\tilde{Q}\}$ ▷ Save the candidate
    $\mathcal{S}_{batch} \leftarrow \mathcal{S}_{best}$
    $\mathcal{S} \leftarrow \mathcal{S} \cup \mathcal{S}_{best}$
  return $\mathcal{S}$

Next, we present our scoring function. To evaluate the quality of the candidate, we consider the following aspects comprehensively:

- **LM Score.** We employ a probabilistic language model (LM) to measure the language fluency of a candidate, which is widely applied in unsupervised text compression and simplification tasks [19, 33]. As the training objective of LMs is to maximize the likelihood of sentences, a fluent sentence would have a higher joint probability, which can be denoted as $f_{LM}(\tilde{Q}) = \ln P_{LM}(\tilde{Q}) = \ln \prod_{i=1}^{T} P(w_i|w_{i-1}, ..., w_1)$, where w_i is i-th token in $\tilde{Q}$ and T is the length of the sentence.
- **Semantic Integrity.** To avoid the dramatic change to the semantics of the original question after edition, we employ cosine similarity to measure the meaning preservation, where the sentence embedding is computed as the weighted average of tokens in sentences. We denote it as $f_{Semantic}(\tilde{Q}) = cos(\tilde{Q}, Q)$.
- **Syntactic Invariance.** Since we would like to ensure the alternative constituents can hold the same syntactic attributes as the original one. This could maintain the syntactic structure of the original question and effectively avoid grammatical confusion. We identify whether the root tags of these constituents are same or not, which can be denoted as $f_{Syntactic}(\tilde{Q}) = \mathbb{I}(Tag_{Q[i]} = Tag_{\tilde{Q}[j]})$. Here $\mathbb{I}(\cdot)$ is an indicator function.

The overall scoring function is the product of the above aspects:

$$f(\tilde{Q}) = f_{LM}(\tilde{Q})^{\alpha} f_{Semantic}(\tilde{Q})^{\beta} f_{Syntactic}(\tilde{Q}), \quad (1)$$

where the weights α and β denote the importance of LM score and semantic integrity, respectively. It is worth that syntactic invariance is a hard indicator function. It only accepts the case when the replaced root tag is unchangeable so there is no importance weight needed. As we can see, $f(\tilde{Q})$ is a scalar that indicates how likely $\tilde{Q}$ can act as a good reasoning question prompt for Q. Eventually, we obtain a set $\mathcal{S}$ that contains k reasoning question prompts.

Prompt Design. With the generated k reasoning question prompts, we construct the prompts for answer generation by concatenating the caption and each reasoning question prompts. Following prior studies on prompt tuning [16, 43, 54], we construct the prompt with the consideration of instruction, context and questions:

Instruction: `Please answer the question according to the contexts.`
Context: `[caption]`.
Question: `[reasoning question prompt]`.
Answer:

We will feed the k prompts into LLMs in turn and greedy decoding on LLMs is performed on each prompt. This results in k candidate answers with their confidence scores.

In Figure 2, different reasoning question prompts capture different objects in the image such as "*clouds and lightning*" and "*this clock*", they can cover possible intents of the original question, which helps LLMs to decode answers with diverse reasoning paths. This strategy has similar principle as Chain-of-Thought [22, 57, 59], which explicates the intermediate reasoning chains of the questions and makes it easier for LLMs to parse the question and do complicated reasoning. After prompting for answer generation, we obtain two candidate answers, that are "*fear*" and "*anxiety*", which correspond to the two reasoning question prompts.

3.3 Prompting for Answer Choosing

Once we obtain multiple candidate answers, we construct prompts to let LLMs choose final answers among these candidate answers, which are known as heuristics-enhanced prompts in Prophet. This facilitates the LLMs to narrow down the range of answers. We follow this strategy but define different confidence scores in Answer Heuristics Construction module.

Answer Heuristics Construction. Starting from the generated candidate answers based on different reasoning question prompts, we define the confidence score of the candidate answer below:

$$P(A) = \sum_{LLM(\tilde{Q})\rightarrow A} P_(\tilde{Q}) P_{LLM}(A|\tilde{Q}), \tag{2}$$

where $P(\tilde{Q})$ is the probability that we generate the $\tilde{Q}$ based on normalized $f(\tilde{Q})$ over k prompts and $P_{LLM}(A|\tilde{Q})$ is the probability of the generated A based on $\tilde{Q}$ via LLMs. Since different reasoning question prompts may lead to the same answer, we can have m candidate answers, where $m \leq k$. As we can see, the confidence score takes both confidences of question edition and answer generation into account, which comprehensively depicts the likelihood of a candidate answer for answering choosing.

Prompt Design. With the generated m candidate answers, we construct the prompts for answer choosing by concatenating the caption, original question and candidate answers:

Instruction: `Please answer the question according to the contexts and candidates.`
Context: `[caption]`.
Question: `[original question]`.
Candidates: `[`A_1 `P(`A_1`)];[`A_2 `P(`A_2`)];...;[`A_m `P(`A_m`)]`
Answer:

where $P(A_m)$ denotes the confidence score for answer A_m, which reminds LLMs to focus more on the candidate answers with higher scores. We consider the answer generated by this prompt as the final answer.

Compared with the two-stage prompting method of Prophet, our method is different in the way of generating and scoring answer candidates, which is rooted in our different motivation. Prophet generates answer candidates by including frequent answers from training set, which is to replay the answer prediction in the training data. Our method generates answer candidates by full-filling the original questions with possible intents, which is to shorten the semantic gap between images and questions under the zero-shot setting. It is worth noting that even though the prompting method is designed for the zero-shot VQA task, we can still insert in-context examples behind the instructional prompt if it is needed.

4 EXPERIMENTS

In this section, we evaluate reasoning question prompts on zero-shot VQA tasks and compare with existing methods. Furthermore, we perform comprehensive analysis to interpret its performance under different scenarios. We also conduce ablation study on important design choices and show some qualitative examples.

4.1 Experimental Setup

Datasets. We evaluate reasoning question prompts on **OK-VQA** [32], **A-OKVQA** [42] and **VQAv2** [14], which contains image-question pairs that are derived from COCO datasets[27]. The questions in these datasets require perception to the image. Some of them even require commonsense beyond the image to answer. Specifically, OK-VQA[1] contains 5,046 test questions. A-OKVQA[2] contains 1,100 and 6,700 questions for validation and testing, respectively. VQAv2[3] is a large dataset, We leverage the validation set of VQAv2 for evaluation, which contains 214,354 questions. For evaluation measurement, we follow their official evaluation metrics to measure the performance.

Comparable Methods. As our reasoning question prompts can collaborate with any LLMs, we evaluate our methods with different LLMs as backbones. Notably, existing methods like PICa and Img2Prompt are prompting methods to provide exemplars prompts for VQA tasks. We consider their methods as baselines then include our reasoning question prompts and observe if there is any performance improvement brought by the involvement of our method.

Besides, we compare our method with other pre-trained zero-shot VQA methods, such as Flamingo [2], Frozen [45] VL-T5 [9], FewVLM [18] and VLKD [10]. These methods aim to propose different pre-trained multi-modal models on large-scale vision-language datasets, which can be easily adapted to new VQA challenges without training.

Implementation Details. For the LM used in the unsupervised question edition module, we leverage a pre-trained LM model from existing work, which is a two-layer, 256 dimensional recurrent neural network with gated recurrent unit (GRU) [23] fine-tuned

[1] https://okvqa.allenai.org/
[2] https://allenai.org/project/a-okvqa/home
[3] https://visualqa.org/download.html

Table 1: Zero-shot evaluation on VQAv2, OK-VQA, and A-OKVQA. The first section contains zero-shot methods with LLMs which utilize no training data but may synthesize some exemplars. The middle section contains zero-shot methods with end-to-end training on other multi-modal data. The last section contains few-shot methods with LLMs. The numbers in brackets denote the improvement gain brought by our reasoning question prompts. The results with ⋄ denote the baselines we implement methods by ourselves. Otherwise, we copy results from their original papers.

Method	Model size	Shot number	Examplar number	OK-VQA test	VQAv2 val	A-OKVQA val	A-OKVQA test
Zero-shot Evaluation with Frozen LLMs							
PICa $_{\{GPT-3\}}$	175B	0	0	17.7	–	23.8$^{\diamond}$	–
Img2Prompt $_{\{OPT\}}$	6.7B	0	30	38.2	52.2$^{\diamond}$	33.3	32.2
Img2Prompt $_{\{OPT\}}$	30B	0	30	41.8	54.2$^{\diamond}$	36.9	33.0
Img2Prompt $_{\{GPT-3\}}$	175B	0	30	42.8	–	38.9$^{\diamond}$	43.4$^{\diamond}$
Img2Prompt $_{\{OPT\}}$	175B	0	30	45.6	**60.6**	42.9	40.7
PICa+RQ prompt $_{\{GPT-3\}}$ (Ours)	175B	0	0	20.3(↑ 2.6)	–	29.0(↑ 5.2)	–
Img2Prompt+RQ prompt $_{\{OPT\}}$ (Ours)	6.7B	0	30	38.5(↑ 0.3)	52.9(↑ 0.7)	36.3(↑ 3.0)	31.5
Img2Prompt+RQ prompt $_{\{OPT\}}$ (Ours)	30B	0	30	42.1(↑ 0.3)	54.5(↑ 0.3)	38.1(↑ 1.2)	35.2(↑ 3.0)
Img2Prompt+RQ prompt $_{\{GPT-3\}}$ (Ours)	175B	0	30	**46.4**(↑ 3.6)	–	**43.2**(↑ 4.3)	**43.9**(↑ 0.5)
Zero-shot Evaluation with Pre-trained VQA methods							
VL-T5 $_{\{no-vqa\}}$	224M	0	0	5.8	13.5	–	–
FewVLM $_{\{large\}}$	740M	0	0	16.5	47.7	–	–
VLKD $_{\{ViT-L/14\}}$	408M	0	0	13.3	44.5	–	–
Frozen	7B	0	0	5.9	29.5	–	–
Flamingo	80B	0	0	50.6	–	–	–
Few-shot Evaluation with Frozen LLMs							
PICa $_{\{GPT-3\}}$	175B	16	16	46.5	54.3	–	–
Prophet $_{\{GPT-3\}}$	175B	20	20	61.1	–	–	–

on OK-VQA test set. Compared with LMs like BERT and Roberta, this model is enriched with syntactic information, which is more suitable for our method. α and β in Equation (1) are set to 0.3 and 1, respectively. We set ρ as 0.5 to avoid overwhelming question reasoning prompts. If there is a maximum limit number for the generated reasoning question prompts, we sort all candidate reasoning question prompts and select the top-k based on their scores. More details about the unsupervised edition module can be found in Appendix A.1.

Regarding LLMs, to show the generalization capability of our reasoning question prompts, we conduct experiments on different LLMs with different sizes, including open source OPT[4], GPT-3[5] and BLOOM.[6]. Regarding different baselines, such as Img2Prompt[7], PICa[8], we follow their official implementation to convert images into captions via either VinVL-base pre-trained checkpoint[9] or BLIP[10] and generate exemplar prompts via either CLIP[11] or fine-tuned T5-large model[12]. Notably, we implement a light version of Img2Prompt on VQAv2 dataset due to our computation limitation, the details of which can be found in Appendix A.2.

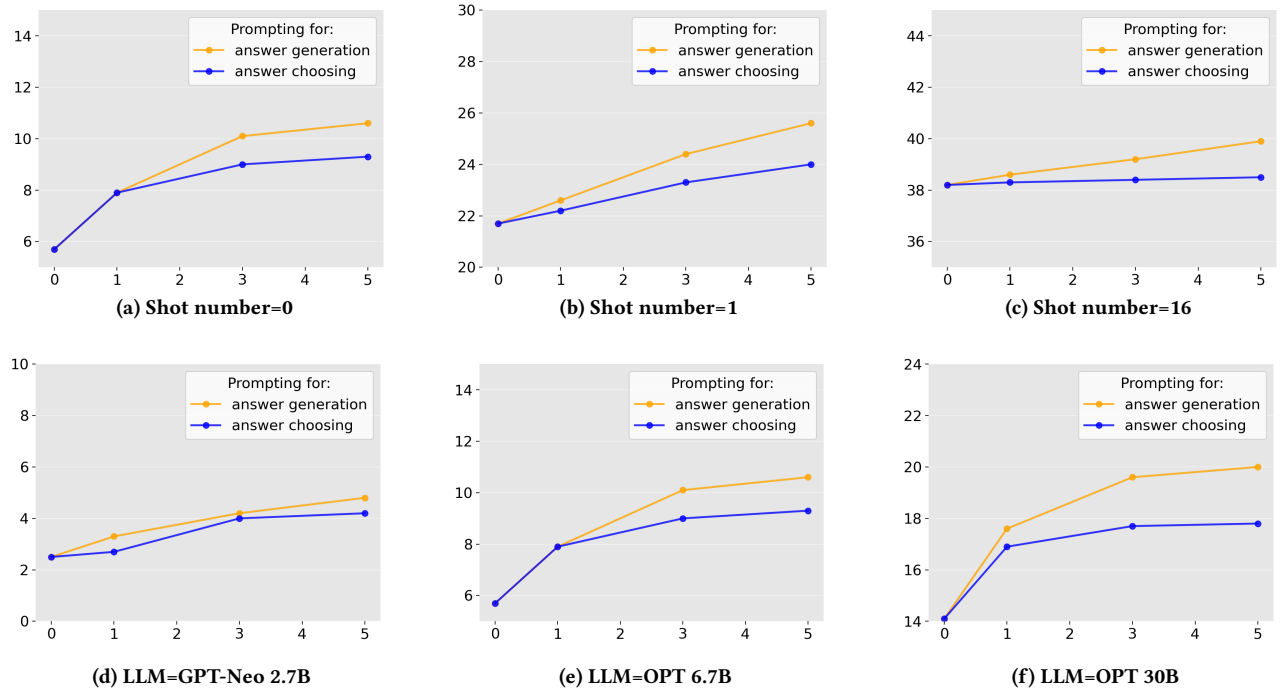
4.2 Main Results

We display our main results in Table 1. We have the following observations based on it:

Overall effect of reasoning question prompts. Our reasoning question prompts can improve the performance of zero-shot VQA methods on most baselines. The absolute improvement ranges from 0.3 to 5.2 points. The largest gain is on A-OKVQA validation set with PICa baseline, where the absolute improvement is 5.2 points. This is a setting without any exemplar, which indicates the potential of reasoning question prompts under the scenarios with no access to any VQA data. Even with some exemplars which are synthetically generated, reasoning question prompts can still improve the zero-shot VQA methods via LLMs. As we can see, even with some syntactic exemplars generated by Img2Prompt model, there is general improvement from reasoning question prompts. We observe the similar effect of RQ prompts with different LLMs, the results of which are displayed in Appendix A.3.

Comparison with other methods. Compared with existing zero-shot methods, reasoning question prompts with Img2Prompt $_{\{GPT-175B\}}$ baseline can outperform all the existing zero-shot VQA methods and achieve the new state-of-the-art results on zero-shot evaluation with frozen LLMs on three out of four data sets. Even

[4]https://huggingface.co/docs/transformers/model_doc/opt
[5]https://openai.com
[6]https://huggingface.co/docs/transformers/model_doc/bloom
[7]https://github.com/salesforce/LAVIS/tree/main/projects/img2llm-vqa
[8]https://github.com/microsoft/PICa
[9]https://github.com/pzzhang/VinVL
[10]https://github.com/salesforce/BLIP
[11]https://github.com/OpenAI/CLIP
[12]https://github.com/google-research/text-to-text-transfer-transformer



(a) Shot number=0 (b) Shot number=1 (c) Shot number=16

(d) LLM=GPT-Neo 2.7B (e) LLM=OPT 6.7B (f) LLM=OPT 30B

Figure 4: (a)-(c) denote evaluation of RQ prompts on the test set of OK-VQA having OPT 6.7B as the LLMs but with different shot numbers. (d)-(f) denote evaluation of RQ prompts on the test set of OK-VQA having shot numbers equal to 0 but with different LLMs. We display results of prompting for answer generation and answer choosing. X-axis denotes the value of k and y-axis denotes the accuracy. When it comes to prompting for answer generation, we report the maximum accuracy among all the reasoning question prompts.

though we cannot defeat Img2Prompt$_{\{\text{OPT-175B}\}}$ on VQAv2 validation set, our reasoning question prompts can still bring in perfromance gain on our light re-implmentation results. We notice that there are some competitive comparable methods like pre-trained VQA method Flamingo$_{\{80B\}}$ and few-shot LLMs-based method Prophet$_{\{\text{GPT-175B}\}}$, which lead to higher results than ours. The former method is computationally expensive, which is pre-trained on billion-scale multi-modal datasets. The latter one makes use of VQA training samples, which can obtain more guidance directly from the training data.

4.3 More Analysis of RQ Prompts

Effect of k for RQ Prompts on Different Shot Numbers. As the results in Table 1 have the mixed effect of shot number and LLMs, to analysis the effect of reasoning question prompts on different shot numbers, we control the other settings unchangeable and see how the performance changes with the increasing number of shot. The results are displayed in Table 2 (a)-(c). As we can see, the performance gradually improves with the increasing k. During answer generation, the more reasoning question prompts generated, the more likely that we can recall the correct answers. We can observe the largest performance gain on the zero-shot setting. This indicates that a reasoning question prompt is more likely to help when the guidance to the question is little. Providing LLMs with self-contained questions, which explicate the intermediate reasoning to LLMs, can fully activate the potentials of LLMs. When the shot number becomes 16, the gain from reasoning question prompts becomes least visible.

Effect of k for RQ Prompts on Different LLMs. We further control the shot number as 0 and test the effect of k for reasoning question prompts on LLMs with different parameter sizes. The results are displayed in Table 2 (d)-(f). Similarly, the performance gain increases with the increasing k. Furthermore, with the increase of model size, the performance gain becomes large. Regarding GPT-Neo-2.7B, the performance increase brought by reasoning question prompts is not so obvious, which only has around 1 point improvement. Regarding OPT 30B, the performance increase brought by reasoning question prompts becomes around 3 points. This is because larger LLMs usually contain more knowledge to answer a question. After implicit intent is resolved by the reasoning question prompts, we can take full advantage of its knowledge to answer questions correctly.

Ablation Study. We further evaluate the performance on different prompt design strategies and the results are displayed in Table 2. If we eliminate the two-stage prompting, and simply choose the answer with highest $P(A)$ in Equation (2) as the final answer. We have around 1 point drop. This indicates that the step of answer choosing is needed. It provides LLMs a chance to review the original question with the consideration of candidate answers. During prompting for

Caption: *This is a blow dryer in a bathroom.*

Question: *What is the appliance the woman is holding used for?*
GT Answer: *drying hair*
Original Answer: *cutting hair*

Prompting for Answer Generation:
Q1: *What is the appliance* ***a blow dryer*** *used for?* $P(\tilde{Q}) = 0.21$
A1: ***drying hair*** $P_{LLM}(A|\tilde{Q}) = 0.15$
Q2: *What is the appliance* ***a bathroom*** *is holding used for?* $P(\tilde{Q}) = 0.29$
A2: ***drying hair*** $P_{LLM}(A|\tilde{Q}) = 0.10$

Prompting for Answer Choosing:
Question: *What is the appliance the woman is holding used for?*
Candidates: *drying hair* (1.00)
Predicted Answer: ***drying hair***

(*a*)

Caption: *A little girl holding a cup with rice in dishes in front of her*

Question: *What is the child eating?*
GT Answer: *rice*
Original Answer: *spaghetti*

Prompting for Answer Generation:
Q1: *what is* ***dishes in front of her****?* $P(\tilde{Q}) = 0.15$
A1: ***rice*** $P_{LLM}(A|\tilde{Q}) = 0.20$
Q2: *What is the child eating?* $P(\tilde{Q}) = 0.7$
A2: *spaghetti* $P_{LLM}(A|\tilde{Q}) = 0.10$
Q3: *what is* ***a cup with food in dishes in front of her****?* $P(\tilde{Q}) = 0.15$
A3: ***rice*** $P_{LLM}(A|\tilde{Q}) = 0.30$

Prompting for Answer Choosing:
Question: *What is the child eating?*
Candidates: *spaghetti (0.48); rice (0.51)*
Predicted Answer: ***rice***

(*b*)

Figure 5: Examples in A-OKVQA validation set the prediction of which are incorrect originally but correct with RQ prompts.

Table 2: Performance on A-OKVQA validation set having Img2Prompt as baselines but with different prompt designs.

Methods	A-OKVQA val
Img2Prompt+QR prompt$_{\{\text{GPT-3 175B}\}}$	43.2
w/o Two-stage prompting	42.2
Prompting for Answer Generation	
w/o LM score	40.5
w/o Semantic integrity	40.8
w/o Syntactic invariance	40.1
Prompting for Answer Choosing	
w Plain answer heuristics	42.0
w/o Candidate construction	42.8

answer generation, we omit the aspects of scoring function in turn, the results indicate that all the aspects are important for generating a reasoning question prompt. Among them, syntactic invariance is most significant aspect which measures the consistency of the substitution segments. A replaced constituent with a different syntactic tagging easily leads to a chaotic sentence that cannot be understood by LLMs. LM score and semantic integrity are also helpful in terms of measuring the sentence fluency and semantic integrity of the sentences. During prompting for answer choosing, we omit the candidate construction and simply include the candidate answer without their confidence scores, which results in a performance drop. After changing the confidence score to $P_{LLM}(A|\tilde{Q})$, the performance decreases, which indicates the importance of our answer heuristics construction.

Case Study. We display some cases in Figure 5 to investigate how our reasoning question prompts work in zero-shot VQA tasks.

Example (a) contains an image of a blow dryer, the generated caption is "*This is a blow dryer in a bathroom*". The visual question is "*What is the appliance the woman is holding used for?*". As we can see, this is an ill-posed question as there is no woman shown on the image. The result of LLMs without any reasoning question prompt is "*cutting hair*", which may caused by the unexpected bias of the LLMs. Based on the caption and question, we generate reasoning question prompts such as "*What is the appliance a blow dryer used for?*" and "*What is the appliance a bathroom is holding used for?*", which successfully bridge the gap between the image and the question, thus LLMs can predict correct answer. Similarly, in example (b), there is a gap between "*the child eating*" and the image. The queried objective is not explicitly mentioned in the question, so LLMs must infer the object that the question is asking about. Reasoning question prompts such as "*What is dishes in front of her?*" and "*What is a cup with food in dishes in front of her?*" explicate the queried object so that LLMs can easily understand the question and return the correct answers. More cases can be found in Appendix A.4.

5 CONCLUSION

In this paper, we investigate zero-shot VQA tasks via LLMs, where images are first converted into captions then LLMs answer questions based on the caption contents. We propose a way to generate reasoning question prompts, which can help explicate the intermediate reasoning step of a question and eliminate the semantic gap between the question and the caption. The experiments show that reasoning question prompts improve existing zero-shot VQA methods with different LLM backbones and achieve a new state-of-the-art performance on multiple zero-shot VQA data sets.

ACKNOWLEDGEMENTS

The authors would like to thank the anonymous reviewers for their insightful comments. This work was supported by the Natural Science Foundation of China (Project No. 62206097) and Shanghai Pujiang Talent Program (Project No. 22PJ1403000).

REFERENCES

[1] Aishwarya Agrawal, Dhruv Batra, Devi Parikh, and Aniruddha Kembhavi. 2018. Don't just assume; look and answer: Overcoming priors for visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 4971–4980.

[2] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, Roman Ring, Eliza Rutherford, Serkan Cabi, Tengda Han, Zhitao Gong, Sina Samangooei, Marianne Monteiro, Jacob Menick, Sebastian Borgeaud, Andrew Brock, Aida Nematzadeh, Sahand Sharifzadeh, Mikolaj Binkowski, Ricardo Barreira, Oriol Vinyals, Andrew Zisserman, and Karen Simonyan. 2022. Flamingo: a Visual Language Model for Few-Shot Learning. In *Advances in Neural Information Processing Systems*. 23716–23736.

[3] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. 2015. Vqa: Visual question answering. In *Proceedings of the IEEE international conference on computer vision*. 2425–2433.

[4] Pratyay Banerjee, Tejas Gokhale, Yezhou Yang, and Chitta Baral. 2020. WeaQA: Weak supervision via captions for visual question answering. *arXiv preprint arXiv:2012.02356* (2020).

[5] Sid Black, Stella Biderman, Eric Hallahan, Quentin Anthony, Leo Gao, Laurence Golding, Horace He, Connor Leahy, Kyle McDonell, Jason Phang, Michael Pieler, USVSN Sai Prashanth, Shivanshu Purohit, Laria Reynolds, Jonathan Tow, Ben Wang, and Samuel Weinbach. 2022. GPT-NeoX-20B: An Open-Source Autoregressive Language Model. (2022). arXiv:2204.06745 [cs.CL]

[6] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems* (2020), 1877–1901.

[7] Soravit Changpinyo, Doron Kukliansky, Idan Szpektor, Xi Chen, Nan Ding, and Radu Soricut. 2022. All you may need for VQA are image captions. *arXiv preprint arXiv:2205.01883* (2022).

[8] Zhenfang Chen, Qinhong Zhou, Yikang Shen, Yining Hong, Hao Zhang, and Chuang Gan. 2023. See, Think, Confirm: Interactive Prompting Between Vision and Language Models for Knowledge-based Visual Reasoning. *arXiv preprint arXiv:2301.05226* (2023).

[9] Jaemin Cho, Jie Lei, Hao Tan, and Mohit Bansal. 2021. Unifying Vision-and-Language Tasks via Text Generation. In *Proceedings of the 38th International Conference on Machine Learning*. 1931–1942.

[10] Wenliang Dai, Lu Hou, Lifeng Shang, Xin Jiang, Qun Liu, and Pascale Fung. 2022. Enabling Multimodal Generation on CLIP via Vision-Language Knowledge Distillation. In *Findings of the Association for Computational Linguistics: ACL 2022*. 2383–2395.

[11] Feng Gao, Qing Ping, Govind Thattai, Aishwarya Reganti, Ying Nian Wu, and Prem Natarajan. 2022. Transform-retrieve-generate: Natural language-centric outside-knowledge visual question answering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 5067–5077.

[12] Diego Garcia-Olano, Yasumasa Onoe, and Joydeep Ghosh. 2022. Improving and diagnosing knowledge-based visual question answering via entity enhanced knowledge injection. In *Companion Proceedings of the Web Conference 2022*. 705–715.

[13] François Gardères, Maryam Ziaeefard, Baptiste Abeloos, and Freddy Lecue. 2020. Conceptbert: Concept-aware representation for visual question answering. In *Findings of the Association for Computational Linguistics: EMNLP 2020*. 489–498.

[14] Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. 2017. Making the V in VQA Matter: Elevating the Role of Image Understanding in Visual Question Answering. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 6325–6334.

[15] Liangke Gui, Borui Wang, Qiuyuan Huang, Alex Hauptmann, Yonatan Bisk, and Jianfeng Gao. 2021. Kat: A knowledge augmented transformer for vision-and-language. *arXiv preprint arXiv:2112.08614* (2021).

[16] Jiaxian Guo, Junnan Li, Dongxu Li, Anthony Tiong, Boyang Li, Dacheng Tao, and Steven Hoi. 2023. From Images to Textual Prompts: Zero-shot Visual Question Answering with Frozen Large Language Models. In *The IEEE/CVF Computer Vision and Pattern Recognition Conference*.

[17] Jingjing Jiang and Nanning Zheng. 2023. MixPHM: Redundancy-Aware Parameter-Efficient Tuning for Low-Resource Visual Question Answering. *arXiv preprint arXiv:2303.01239* (2023).

[18] Woojeong Jin, Yu Cheng, Yelong Shen, Weizhu Chen, and Xiang Ren. 2022. A Good Prompt Is Worth Millions of Parameters: Low-resource Prompt-based Learning for Vision-Language Models. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 2763–2775.

[19] Katharina Kann, Sascha Rothe, and Katja Filippova. 2018. Sentence-Level Fluency Evaluation: References Help, But Can Be Spared!. In *Proceedings of the 22nd Conference on Computational Natural Language Learning*. 313–323.

[20] Junyeob Kim, Hyuhng Joon Kim, Hyunsoo Cho, Hwiyeol Jo, Sang-Woo Lee, Sang-goo Lee, Kang Min Yoo, and Taeuk Kim. 2022. Ground-truth labels matter: A deeper look into input-label demonstrations. *arXiv preprint arXiv:2205.12685* (2022).

[21] Hannah Rose Kirk, Yennie Jun, Filippo Volpin, Haider Iqbal, Elias Benussi, Frederic Dreyer, Aleksandar Shtedritski, and Yuki Asano. 2021. Bias out-of-the-box: An empirical analysis of intersectional occupational biases in popular generative language models. *Advances in neural information processing systems* (2021), 2611–2624.

[22] Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. *arXiv preprint arXiv:2205.11916* (2022).

[23] Dhruv Kumar, Lili Mou, Lukasz Golab, and Olga Vechtomova. 2020. Iterative Edit-Based Unsupervised Sentence Simplification. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. 7918–7928.

[24] Mingxiao Li and Marie-Francine Moens. 2022. Dynamic Key-Value Memory Enhanced Multi-Step Graph Reasoning for Knowledge-Based Visual Question Answering. In *Proceedings of the AAAI Conference on Artificial Intelligence*. 10983–10992.

[25] Xiang Lisa Li and Percy Liang. 2021. Prefix-Tuning: Optimizing Continuous Prompts for Generation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. 4582–4597.

[26] Sheng Liang, Mengjie Zhao, and Hinrich Schütze. 2022. Modular and Parameter-Efficient Multimodal Fusion with Prompting. *arXiv preprint arXiv:2203.08055* (2022).

[27] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. 2014. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference*. 740–755.

[28] Yuanze Lin, Yujia Xie, Dongdong Chen, Yichong Xu, Chenguang Zhu, and Lu Yuan. 2022. Revive: Regional visual representation matters in knowledge-based visual question answering. *arXiv preprint arXiv:2206.01201* (2022).

[29] Jiachang Liu, Dinghan Shen, Yizhe Zhang, Bill Dolan, Lawrence Carin, and Weizhu Chen. 2021. What Makes Good In-Context Examples for GPT-3? *arXiv preprint arXiv:2101.06804* (2021).

[30] Oscar Mañas, Pau Rodriguez, Saba Ahmadi, Aida Nematzadeh, Yash Goyal, and Aishwarya Agrawal. 2022. MAPL: Parameter-Efficient Adaptation of Unimodal Pre-Trained Models for Vision-Language Few-Shot Prompting. *arXiv preprint arXiv:2210.07179* (2022).

[31] Kenneth Marino, Xinlei Chen, Devi Parikh, Abhinav Gupta, and Marcus Rohrbach. 2021. Krisp: Integrating implicit and symbolic knowledge for open-domain knowledge-based vqa. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 14111–14121.

[32] Kenneth Marino, Mohammad Rastegari, Ali Farhadi, and Roozbeh Mottaghi. 2019. Ok-vqa: A visual question answering benchmark requiring external knowledge. In *Proceedings of the IEEE/cvf conference on computer vision and pattern recognition*. 3195–3204.

[33] Ning Miao, Hao Zhou, Lili Mou, Rui Yan, and Lei Li. 2019. CGMH: Constrained Sentence Generation by Metropolis-Hastings Sampling. In *Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence and Thirty-First Innovative Applications of Artificial Intelligence Conference and Ninth AAAI Symposium on Educational Advances in Artificial Intelligence*. 6834–6842.

[34] Sewon Min, Xinxi Lyu, Ari Holtzman, Mikel Artetxe, Mike Lewis, Hannaneh Hajishirzi, and Luke Zettlemoyer. 2022. Rethinking the Role of Demonstrations: What Makes In-Context Learning Work? *arXiv preprint arXiv:2202.12837* (2022).

[35] Swaroop Mishra, Daniel Khashabi, Chitta Baral, and Hannaneh Hajishirzi. 2022. Cross-task generalization via natural language crowdsourcing instructions. In *60th Annual Meeting of the Association for Computational Linguistics*.

[36] Medhini Narasimhan, Svetlana Lazebnik, and Alexander Schwing. 2018. Out of the Box: Reasoning with Graph Convolution Nets for Factual Visual Question Answering. In *Advances in Neural Information Processing Systems*.

[37] Guanghui Qin and Jason Eisner. 2021. Learning How to Ask: Querying LMs with Mixtures of Soft Prompts. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. 5203–5212.

[38] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*. 8748–8763.

[39] Teven Le Scao, Thomas Wang, Daniel Hesslow, Lucile Saulnier, Stas Bekman, M Saiful Bari, Stella Bideman, Hady Elsahar, Niklas Muennighoff, Jason Phang, et al. 2022. What Language Model to Train if You Have One Million GPU Hours? *arXiv preprint arXiv: 2210.15424* (2022).

[40] Timo Schick and Hinrich Schütze. 2020. Exploiting cloze questions for few shot text classification and natural language inference. *arXiv preprint arXiv:2001.07676* (2020).

[41] Patrick Schramowski, Cigdem Turan, Nico Andersen, Constantin A. Rothkopf, and Kristian Kersting. 2022. Large pre-trained language models contain human-like biases of what is right and wrong to do. *Nature Machine Intelligence* (2022), 258–268.

[42] Dustin Schwenk, Apoorv Khandelwal, Christopher Clark, Kenneth Marino, and Roozbeh Mottaghi. 2022. A-okvqa: A benchmark for visual question answering using world knowledge. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part VIII*. 146–162.

[43] Zhenwei Shao, Zhou Yu, Meng Wang, and Jun Yu. 2023. Prompting Large Language Models with Answer Heuristics for Knowledge-based Visual Question Answering. *arXiv preprint arXiv:2303.01903* (2023).

[44] Haoyu Song, Li Dong, Weinan Zhang, Ting Liu, and Furu Wei. 2022. CLIP Models are Few-Shot Learners: Empirical Studies on VQA and Visual Entailment. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 6088–6100.

[45] Maria Tsimpoukelli, Jacob Menick, Serkan Cabi, S. M. Ali Eslami, Oriol Vinyals, and Felix Hill. 2021. Multimodal Few-Shot Learning with Frozen Language Models. In *Advances in Neural Information Processing Systems*. 200–212.

[46] Maria Tsimpoukelli, Jacob L Menick, Serkan Cabi, SM Eslami, Oriol Vinyals, and Felix Hill. 2021. Multimodal few-shot learning with frozen language models. *Advances in Neural Information Processing Systems* (2021), 200–212.

[47] Ben Wang and Aran Komatsuzaki. 2021. GPT-J-6B: A 6 Billion Parameter Autoregressive Language Model. *github* (2021).

[48] Peng Wang, Qi Wu, Chunhua Shen, Anton van den Hengel, and Anthony Dick. 2015. Explicit knowledge-based reasoning for visual question answering. *arXiv preprint arXiv:1511.02570* (2015).

[49] Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, and Denny Zhou. 2022. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171* (2022).

[50] Yizhong Wang, Swaroop Mishra, Pegah Alipoormolabashi, Yeganeh Kordi, Amirreza Mirzaei, Anjana Arunkumar, Arjun Ashok, Arut Selvan Dhanasekaran, Atharva Naik, David Stap, et al. 2022. Super-NaturalInstructions:Generalization via Declarative Instructions on 1600+ Tasks. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*. 5085–5109.

[51] Jason Wei, Maarten Bosma, Vincent Y Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. 2021. Finetuned language models are zero-shot learners. *arXiv preprint arXiv:2109.01652* (2021).

[52] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Ed Chi, Quoc Le, and Denny Zhou. 2022. Chain of thought prompting elicits reasoning in large language models. *arXiv preprint arXiv:2201.11903* (2022).

[53] Jialin Wu, Jiasen Lu, Ashish Sabharwal, and Roozbeh Mottaghi. 2022. Multi-modal answer validation for knowledge-based vqa. In *Proceedings of the AAAI Conference on Artificial Intelligence*. 2712–2721.

[54] Zhengyuan Yang, Zhe Gan, Jianfeng Wang, Xiaowei Hu, Yumao Lu, Zicheng Liu, and Lijuan Wang. 2022. An Empirical Study of GPT-3 for Few-Shot Knowledge-Based VQA. In *The Thirty-Sixth AAAI Conference on Artificial Intelligence*. 3081–3089.

[55] Pengchuan Zhang, Xiujun Li, Xiaowei Hu, Jianwei Yang, Lei Zhang, Lijuan Wang, Yejin Choi, and Jianfeng Gao. 2021. VinVL: Revisiting Visual Representations in Vision-Language Models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 5579–5588.

[56] Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, Todor Mihaylov, Myle Ott, Sam Shleifer, Kurt Shuster, Daniel Simig, Punit Singh Koura, Anjali Sridhar, Tianlu Wang, and Luke Zettlemoyer. 2022. OPT: Open Pre-trained Transformer Language Models. *arXiv preprint arXiv:2205.01068* (2022).

[57] Zhuosheng Zhang, Aston Zhang, Mu Li, and Alex Smola. 2022. Automatic chain of thought prompting in large language models. *arXiv preprint arXiv:2210.03493* (2022).

[58] Zihao Zhao, Eric Wallace, Shi Feng, Dan Klein, and Sameer Singh. 2021. Calibrate before use: Improving few-shot performance of language models. In *International Conference on Machine Learning*. 12697–12706.

[59] Denny Zhou, Nathanael Schärli, Le Hou, Jason Wei, Nathan Scales, Xuezhi Wang, Dale Schuurmans, Olivier Bousquet, Quoc Le, and Ed Chi. 2022. Least-to-most prompting enables complex reasoning in large language models. *arXiv preprint arXiv:2205.10625* (2022).

[60] Zihao Zhu, Jing Yu, Yujing Wang, Yajing Sun, Yue Hu, and Qi Wu. 2020. Mucko: Multi-Layer Cross-Modal Knowledge Reasoning for Fact-based Visual Question Answering. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*. 1097–1103.

[61] Maryam Ziaeefard and Freddy Lecue. 2020. Towards Knowledge-Augmented Visual Question Answering. In *Proceedings of the 28th International Conference on Computational Linguistics*. 1863–1873.

A APPENDIX

A.1 Details about Unsupervised Question Edition

For each sentence, we use CoreNLP[13] to construct the constituency tree and Spacy[14] to obtain the part-of-speech and dependency tags of the words.

The syntax-aware LM model we used takes words, POS tags and dependency tags as the input, which can be denoted as:

$$\mathbf{w} = [\mathbf{v}(w); \mathbf{p}(w); \mathbf{d}(w)],$$

where $\mathbf{v}(w)$ is the word embeding, $\mathbf{p}(w)$ is the POS tag embedding and $\mathbf{d}(w)$ is the dependency tag embedding. The dimensions of POS tag and dependency tag are 150, and the dimension of word embedding is 300. $\mathbf{w}$ is fed into the LM [23], which enables a LM to be sensitive to the sentence structure. We directly take the checkpoint[15] of the syntax-aware LM model in prior study [23] on text simplification as the initialization. This checkpoint is initially trained on WikiLarge datasets. We fine-tune the model with questions in OK-VQA test data, so that the LM model can be quickly adapted to VQA domains. Regarding fine-tuning, we use the Stochastic Gradient Descent algorithm with 0.4 as the dropout rate and 32 as the batch size.

A.2 Details of Re-implementation of Img2Prompt on VQAv2 Dataset

We re-implement the source code of Img2Prompt to generate synthetic examples. Due to the large size of the VQAv2 dataset and our limited computational resource, we implement a light version. Specifically, Img2prompt leverages BLIP to generate captions from a given image and conduct image-question matching. In the official implementation setting, they sample 10 image patches and then generate 100 question-relevant captions, from which they can produce 30 question-answer pairs. For us, we sample 10 image patches for each image but simply generate 20 captions by adjusting the number of the generation. Based on these 20 captions, we subsequently generate 10 question-answer pairs. These question-answer pairs are utilized as the exemplar prompts for answer generation and answer choosing. Therefore, there might be some information loss in our implementation as some important exemplars might be filtered out.

A.3 RQ Prompts with Different LLMs

To verify the scaling effect of reasoning question prompts with different LLMs, we conduct experiments on A-OKVQA validation set having Img2Prompt as baselines but with different LLMs. The result is displayed in Table 3. Specifically, we evaluate on GPT-3.5 175*B*, GPT-Neo 2.7B [5], BLOOM 7.1B [39], GPT-J 6B [47] and OPT-125M [56]. As we can see, the performance of zeros-shot VQA tasks is affected by the size of LLMs. A LLM with larger model size usually results in a better performance, which is also verified in prior paper [16]. Importantly, including reasoning question prompts can always improve the performance, which further verifies the generalization capability of our method.

Table 3: Zero-shot performance A-OKVQA validation set having Img2Prompt as baselines but with different LLMs. Δ denotes the performance gain brought by QR prompts.

LLMs	Img2Prompt	+QR prompt	Δ
GPT-3 175B	38.9	43.2	↑ 4.3
GPT-3.5 175B	37.1	40.3	↑ 3.2
GPT-Neo 2.7B	29.7	31.5	↑ 1.8
BLOOM 7.1B	29.8	32.1	↑ 2.3
GPT-J 6B	32.5	33.1	↑ 0.6
OPT 125M	10.8	13.3	↑ 2.5

A.4 Case Study

We display more examples in OK-VQA test set to show how the reasoning question prompts work in zero-shot VQA tasks. The displayed examples are predicted by Img2Prompt+QR prompt $_{\{\text{OPT-30B}\}}$ and the original predictions without QR prompts are incorrect. In the Figure, the edited segments of the question are highlighted with red color and the correct predicted answers are highlighted with green color.

Caption: *A child's bedroom with pink and white decor.*

Question: *Is this a room for a boy or girl?*
GT answer: *girl*

Prompting for Answer Generation:
Q1: *Is pink and white decor a room for a boy or girl?*
A1: *It is room for boy.*
Q2: *Is this pink and white decor for a boy or girl?*
A2: *girl*
Q3: *Is this a room for pink and white decor?*
A3 : *no*
Prompting for Answer Choosing:
Question: *Is this a room for a boy or girl?*
Candidates: *girl(0.51); it is room for boy(0.21); no(0.27)*
Predicted Answer: *girl*

Example 1

[13] https://stanfordnlp.github.io/CoreNLP/
[14] https://spacy.io/
[15] https://github.com/ddhruvkr/Edit-Unsup-TS

Caption: *A lone park bench sitting in the middle of a lush green park.*

Question: *What is the small white outdoor house like building called?*
GT answer: *gazebo; shed; casida*

Prompting for Answer Generation:
Q1: *What is the small white outdoor house like the middle of a lush green park called?*
A1: *greenhouse*
Q2: *What is the small white outdoor house like the middle called?*
A2: *gazebo*
Prompting for Answer Choosing:
Question: *What is the small white outdoor house like building called?*
Candidates: *gazebo(0.25); greenhouse(0.74)*
Predicted Answer: *gazebo*

Example 2

Caption: *A worker takes a stoplight down from the pole.*

Question: *What could this guy be doing?*
GT answer: *fixing traffic light; working; fixing stoplight*

Prompting for Answer Generation:
Q1: *what could a worker be doing?*
A1: *working*
Q2: *what could a stoplight be doing?*
A2: *working*
Q3: *What could this guy be doing?*
A3: *he could be traffic light operator*
Prompting for Answer Choosing:
Question: *What could this guy be doing?*
Candidates: *working(0.89); he could be traffic light operator(0.11)*
Predicted Answer: *working*

Example 3

Caption: *An airplane parked on the tarmac at the airport.*

Question: *What cartoon is on the side of the plane?*
GT answer: *hello kitty; bart Simpson; mickey mouse*

Prompting for Answer Generation:
Q1: *what cartoon is on the side of an airport?*
A1: *hello kitty*
Q2: *what cartoon is on an airplane of the plane?*
A2: *hello kitty*
Q3: *What cartoon is on the side of the plane?*
A3: *cartoon is of man in suit and tie*
Prompting for Answer Choosing:
Question: *What cartoon is on the side of the plane?*
Candidates: *hello kitty(0.99); cartoon is of man in suit and tie(0.01)*
Predicted Answer: *hello kitty*

Example 4

Caption: *A clock on a building next to a building with a "Bart" logo.*

Question: *What is the purpose of the white circle?*
GT answer: *clock*

Prompting for Answer Generation:
Q1: *what is a clock of the white circle?*
A1: *clock*
Q2: *what is the purpose of a building with a "bart" logo?*
A2: *clock*
Q3: *What is the purpose of the white circle?*
A3: *white circle is sun dial*
Prompting for Answer Choosing:
Question: *What is the purpose of the white circle?*
Candidates: *white circle is sun dial(0.09); clock(0.91)*
Predicted Answer: *clock*

Example 5