Generate then Select: Open-ended Visual Question Answering Guided by World Knowledge

Xingyu Fu1∗, Sheng Zhang2, Gukyeong Kwon2, Pramuditha Perera2, Henghui Zhu2, Yuhao Zhang2, Alexander Hanbo Li2, William Wang2, Zhiguo Wang2, Vittorio Castelli2, Patrick Ng2, Dan Roth2, Bing Xiang2

1 University of Pennsylvania, 2 AWS AI Labs
xingyuf2@seas.upenn.edu, {zshe, gukyeong, pramudi, henghui, yhzhang, hanboli, wyw, zhiguow, vittorca, patricng, drot, bxiang}@amazon.com

Abstract

The open-ended Visual Question Answering (VQA) task requires AI models to jointly reason over visual and natural language inputs using world knowledge. Recently, pre-trained Language Models (PLM) such as GPT-3 have been applied to the task and shown to be powerful world knowledge sources. However, these methods suffer from low knowledge coverage caused by PLM bias – the tendency to generate certain tokens over other tokens regardless of prompt changes, and high dependency on the PLM quality – only models using GPT-3 can achieve the best result.

To address the aforementioned challenges, we propose RASO: a new VQA pipeline that deploys a generate-then-select strategy guided by world knowledge for the first time. Rather than following the de facto standard to train a multi-modal model that directly generates the VQA answer, RASO first adopts PLM to generate all the possible answers, and then trains a lightweight answer selection model for the correct answer. As proved in our analysis, RASO expands the knowledge coverage from in-domain training data by a large margin. We provide extensive experimentation and show the effectiveness of our pipeline by advancing the state-of-the-art by +4.1% on OK-VQA, without additional computation cost. Code and models are released at http://cogcomp.org/page/publication_view/1010

1 Introduction

Open-ended Visual Question Answering (VQA), that requires answering a question based on an image, has received much attention in machine learning research in the past decade (Antol et al., 2015; Goyal et al., 2017). Knowledge-based VQA (Marino et al., 2019; Schwenk et al., 2022) is a variant of VQA, where models have to use external knowledge that is not present in the image to generate the answer. It is a more challenging problem as it requires joint reasoning over visual and natural language inputs using world knowledge. For example, in Figure 1, the VQA model needs to conduct multiple levels of inference: to detect the objects in the image (e.g. laptops, whiteboard, etc), to retrieve external world knowledge (e.g. university is an institution and has lecture rooms, lecture rooms have laptops, stairs, and whiteboard, etc), and combine the important visual parts with retrieved knowledge to induce the final answer (e.g. university).

In this paper, we focus on improving the important step of external knowledge retrieval. A common procedure of previous VQA methods (Marino et al., 2021; Wu et al., 2022) is to retrieve with knowledge graphs from diverse knowledge bases (e.g. Wikipedia (Wikipedia contributors, 2004), ConceptNet (Liu and Singh, 2004), etc.), with the results being input to an answer generation model. However, the retrieved knowledge could be noisy, irrelevant, and redundant, and therefore lead to mismatches that limit the VQA performance. Motivated by the development of large-scale PLMs such as GPT-3 (Brown et al., 2020) that obtain state-of-the-art (SOTA) performance in most NLP tasks including text generation (Chowdhery et al., 2022), more recent approaches PiCA (Yang et al., 2022) and KAT (Gui et al., 2022) propose to re-
What kind of institution does this image depict?

Please list all the possible answers to the question:
- Office, or computer lab, or it room
- Middle school, or college, or high school
- What level of schooling are these people in?
- What kind of institution does this image depict?
- What kind of institution does this image depict?

Q: What room is this?
A: Office, or computer lab, or it room

Prompt.Q

Please answer the question according to the above context. List all the possible answers.
- Office, or computer lab, or it room
- What kind of institution does this image depict?
- A large room full of laptops with people in the background, wall, computer ...
- What kind of institution does this image depict?
- What level of schooling are these people in?

Prompt.QC

Table 1: Knowledge coverage (%) of different five PLMs evaluated on OK-VQA. Prompt.Q means that the prompt to PLM is constructed by the VQA question only, and Prompt.QC means that the prompt is constructed by the VQA image and question together. Note that the GPT-3 score is taken from (Yang et al., 2022).

<table>
<thead>
<tr>
<th>Prompt</th>
<th>GPT-J</th>
<th>UL2</th>
<th>GPT-3</th>
<th>OPT</th>
<th>Codex</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q</td>
<td>32.4</td>
<td>32.6</td>
<td>34.21</td>
<td>44.8</td>
<td></td>
</tr>
<tr>
<td>QC</td>
<td>37.1</td>
<td>37.5</td>
<td>48.0</td>
<td>37.8</td>
<td>52.9</td>
</tr>
</tbody>
</table>

Table 1 proves that existing VQA approaches using PLMs can only cover less than half (37% - 53%) of the required external knowledge. Further, the PLM outputs and get the answer choice list. Note that the list is ranked by PLM probability from high to low. More details can be found in Section 3.1.

Figure 2: Our multiple choice generation step. Given an image, we use existing tools to get the caption and object tags. We then select most similar examples from the training data and construct the two prompts. We combine the PLM outputs and get the answer choice list. Note that the list is ranked by PLM probability from high to low. More details can be found in Section 3.1.

Figure 3: Our answer selection step. Before selecting the final answer, we first use the same PLM to generate a chain-of-thought rationale to guide the process. Then input being the image or its caption, the question, CoT rationale, and answer choices from Step 1, we train a model to output the correct answer. See Section 4.4 for details about the answer selection models we experiment with.
small difference (5% - 8%) between $Prompt_Q$ and $Prompt_{QC}$ coverage percentages show that PLM bias – the tendency to generate certain tokens over others given the same question – is not alleviated by prompt changes such as the inclusion of the image information or not.

To address these challenges, we propose RASO, a new VQA pipeline that expands world knowledge retrieval by requesting PLMs to generate multiple answer choices, followed by an answer selection model. As shown in Figure 2, we first propose a new prompting method to retrieve a long list of possible answers using in-context examples from in-domain training data. Note that for the example data in Figure 1, the PiCA end-task output would be “office” as in $A_{QC}$ in Figure 2. With this prompting method, we expand the external knowledge coverage by more than +20% for each PLM, without additional training data. Then, as illustrated in Figure 3, we propose a chain-of-thought (CoT) (Wei et al., 2022) guided answer selection approach. By plugging in the previous SOTA method KAT (Gui et al., 2022) as the answer selector, we achieve the new SOTA performance 58.5% (+4.1%) on the OK-VQA dataset without additional computation effort.

Extensive experiments in Section 4 suggest that RASO provides a general way to increase the retrieved world knowledge coverage using PLMs, boosting end-task performance without additional computation cost. We believe our proposed pipeline motivates a new type of generate-then-select VQA method and facilitates future work.

Our main contributions are: (a) We provide a new prompting method using PLMs that extends the retrieved external knowledge coverage by 20% over previous approaches in VQA; (b) We are the first to propose a general generate-then-select VQA pipeline, different from the de facto tradition of direct generation approaches; (c) We achieve the new SOTA on the challenging OK-VQA benchmark.

2 Related Work

2.1 VQA Methods

Visual question answering (VQA) has always been one of the most popular topics in the natural language and computer vision community over recent years. While the VQA task is free-form and open-ended as first proposed in (Antol et al., 2015), a large portion of previous methods (Shih et al., 2016; Anderson et al., 2018; Lu et al., 2019; Gardères et al., 2020) cast it as a classification problem. It’s a common strategy for them to construct a target vocabulary from the dataset’s training set by answer frequency, resulting in around two to four thousand candidates in the target vocabulary (Ben-Younes et al., 2017; Yu et al., 2019; Marino et al., 2021; Wu et al., 2022). These methods suffer from the limited answer vocabulary – if the gold answer is outside of the vocabulary, then there is no way for these models to have the correct answer.

Rather than closed-set classification, several recent methods focus on direct generating for the correct answer (Gui et al., 2022; Salaberria et al., 2023) using transformer-based models such as T5 (Raffel et al., 2020). Large-scale multi-modal models trained on multiple vision language tasks (Alayrac et al., 2022; Chen et al., 2022) have also become popular and achieved good performance on the OK-VQA dataset. However, these models are not publicly available and necessitate a vast quantity of data and computation resources.

Different from all the previous approaches that are either classification or direct generation, our proposed pipeline RASO is the first approach ever to follow a generate-then-select strategy, as far as this paper is written. We hope to benefit from less computation cost in the selection part compared to direct generation, while keeping the free-form open-ended answer vocabulary from the answer generation part.

2.2 Knowledge-based VQA

While significant progress (Lu et al., 2016; Anderson et al., 2018; Lu et al., 2019; Jiang et al., 2020; Marino et al., 2021; Biten et al., 2022) has been made on the most famous VQA benchmarks (Antol et al., 2015; Goyal et al., 2017; Wang et al., 2017; Singh et al., 2019), researchers start to raise more challenging questions that require external knowledge not inside the image to answer (Marino et al., 2019; Zellers et al., 2019; Park et al., 2020; Schwenk et al., 2022; Fu et al., 2022).

Two-step approaches (Marino et al., 2021; Wu et al., 2022; Gui et al., 2022; Lin and Byrne, 2022; Gao et al., 2022; Hu et al., 2022; Lin et al., 2022) that explicitly retrieve world knowledge as input to the end-task model have received much attention. However, these methods could retrieve noisy and redundant information that limits the VQA performance, or have low knowledge coverage. In contrast, without retrieving documents, they
may suffer from PLM hallucinations. To address these problems, we treat LLM as a world knowledge source with wide coverage, and propose new prompt-engineering methods to retrieve succinct but higher-quality knowledge, represented as answer choices.

3 Method

Our method consists of two steps: answer choices generation and answer selection. The overview of the proposed model is shown in Figures 2 and 3.

Problem Formulation Given a training dataset $D = \{(v_i, q_i, a_i)\}_{i=1}^{N}$, where $v_i$ denotes the i-th training image and $N$ is the total number of the training images, $q_i$ and $a_i$ represent the i-th question and its corresponding answer, respectively. We deploy a generate-then-select strategy to first generate a set of answer choices using a frozen PLM $g$, then trains a model $p$ to select the correct answer from it. $g$ takes $v_i$ and $q_i$ as inputs, and generates all the possible answers $A_i = \{a_{i0}, a_{i1}, a_{i2}, \ldots\}$. Finally, $p$ takes $v_i$, $q_i$, and $A_i$ as inputs and learns a set of parameters $\theta$ to select from $A_i$ for the final answer.

3.1 Answer Choices Generation

We design our generation process with inspirations from the previous work (Yang et al., 2022; Gui et al., 2022). As demonstrated in Figures 2 and 4, we follow a similar strategy to use few-shot in-context learning and leverage a frozen PLM $g$ to generate all the possible answer choices.

For each image-question pair, we first convert the image $v_i$ into a textual context $c_i$ following (Yang et al., 2022), where $c_i$ consists of a caption generated from an image captioning model (Zhang et al., 2021) and a list of tags predicted by the public Microsoft Azure tagging API. We then construct two carefully designed text prompts $\text{Prompt}_Q$ and $\text{Prompt}_{QC}$, where $Q$ stands for question and $QC$ stands for question and context.

$\text{Prompt}_Q$ consists of a general instruction sentence: “Please list all the possible answers to the question.”, the textual context, the question, and few-shot in-context examples. The examples are context-question-answers triples taken from the training set that are most similar to the current image-question pair. Since we want to generate all the possible answers, we use all the gold answers and connect them with “or” in the few-shot examples. $\text{Prompt}_Q$ has similar components: a slightly different instruction sentence, the question, and few-shot examples of question-answers pairs.

Following (Yang et al., 2022; Gui et al., 2022), we use 16-shot in-context examples and calculate the similarity scores using CLIP (Radford et al., 2021) embedding of the images and the questions. We utilize the frozen PLM $g$ to generate outputs for both $\text{Prompt}_Q$ and $\text{Prompt}_{QC}$ as demonstrated in Figure 4. The outputs are combined together to form the final answer choices $A_i = \{a_{i0}, a_{i1}, a_{i2}, \ldots\}$ for the current image-question pair. Our goal is to have $a_i \in A_i$.

3.2 Answer Selection

Given $v_i$, $c_i$, $q_i$, $A_i$, this step trains a model $p$ that selects $\hat{a_i}$ from $A_i$. Our goal is for $p$ to output $a_i$ when $a_i \in A_i$.

Before training $p$, we first generate chain-of-thought (CoT) (Wei et al., 2022) style rationales to help guide the selection process, with inspirations from (Schwenk et al., 2022). Specifically, a fixed prompt is pre-designed to generate CoT rationales, with details in Figure 6 in Appendix A.

We then construct the input for the answer selection model. In this paper, we plug in existing text generation models as $p$, and require them to output one choice with further fine-tuning on OK-VQA. For each image-question pair, we concatenate the question $q_i$, the image – represented by either $c_i$ or the image embedding using CLIP model (Radford et al., 2021), the CoT rationale $\text{cot}_i$, and the generated answers choices $A_i$. We also add sentinel tokens such that the input turns out to be in the following format: $\text{Context}: c_i, \text{question}: q_i, \text{rationale}: \text{cot}_i, \text{choices}: A_i, \text{answers}$: with minor adaptions for each specific $p$. Check Figure 5 for inference.

4 Experiment

4.1 Dataset

OK-VQA (Marino et al., 2021) is a widely used VQA dataset that requires external world knowledge outside of the image to answer the question. The dataset contains 14,031 images from the COCO dataset (Lin et al., 2014) and 14,055 crowd-sourced questions covering a variety of knowledge categories, with 9,009 training data and 5,046 testing data. Each question has ten annotated answers.

---

1 Azure Tagging API: https://westus.dev.cognitive.microsoft.com/docs/services/computer-vision-v3-2/operations/56f91f2e778daf14ad49ff21b
According to the given context, please answer the question by selecting the correct answer.

Context: A large room full of laptops with people in the background. Wall, computer...

Q: What kind of institution does this image depict? (possibly repeated), and we follow the standard evaluation metric recommended by the VQA challenge (Antol et al., 2015). The external knowledge required in OK-VQA is not provided and there is no designated external knowledge source (such as a knowledge base), leaving the benchmark more challenging.

#### Table 2: Answer choices generation result on OK-VQA, representing the external knowledge coverage. Top 1, Top 3, Top 5, and All represent the highest accuracy that can be achieved using top 1, top 3, top 5, and all answer choices. All results are in accuracy scores evaluated following (Antol et al., 2015). “both” means that we combine the answer choices generated using both prompts. “ensembled” means that we combine the answer choices of all four PLMs. Note that the GPT-3 result is taken from (Yang et al., 2022).

<table>
<thead>
<tr>
<th>PLM</th>
<th>Prompt Type</th>
<th>Top1 (%)</th>
<th>Top3 (%)</th>
<th>Top5 (%)</th>
<th>All (%)</th>
<th>Avg #</th>
</tr>
</thead>
<tbody>
<tr>
<td>GPT J</td>
<td>PromptQ</td>
<td>32.4</td>
<td>46.1</td>
<td>46.7</td>
<td>46.7</td>
<td>2.6</td>
</tr>
<tr>
<td></td>
<td>PromptQC</td>
<td>37.1</td>
<td>49.5</td>
<td>50.7</td>
<td>50.7</td>
<td>3.0</td>
</tr>
<tr>
<td></td>
<td>both</td>
<td>37.1</td>
<td>52.0</td>
<td>55.9</td>
<td>57.1</td>
<td>4.1</td>
</tr>
<tr>
<td>UL2</td>
<td>PromptQ</td>
<td>32.6</td>
<td>45.4</td>
<td>46.4</td>
<td>46.5</td>
<td>2.7</td>
</tr>
<tr>
<td></td>
<td>PromptQC</td>
<td>37.5</td>
<td>51.3</td>
<td>52.8</td>
<td>52.9</td>
<td>3.0</td>
</tr>
<tr>
<td></td>
<td>both</td>
<td>37.5</td>
<td>53.1</td>
<td>57.0</td>
<td>58.0</td>
<td>4.1</td>
</tr>
<tr>
<td>GPT-3</td>
<td>PromptQC</td>
<td>48.0</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>OPT</td>
<td>PromptQ</td>
<td>34.21</td>
<td>48.45</td>
<td>49.7</td>
<td>49.8</td>
<td>3.0</td>
</tr>
<tr>
<td></td>
<td>PromptQC</td>
<td>37.8</td>
<td>52.9</td>
<td>55.0</td>
<td>55.4</td>
<td>3.7</td>
</tr>
<tr>
<td></td>
<td>both</td>
<td>37.8</td>
<td>55.6</td>
<td>61.0</td>
<td>63.4</td>
<td>5.2</td>
</tr>
<tr>
<td>Codex</td>
<td>PromptQ</td>
<td>44.8</td>
<td>58.8</td>
<td>59.8</td>
<td>59.8</td>
<td>3.1</td>
</tr>
<tr>
<td></td>
<td>PromptQC</td>
<td>52.9</td>
<td>67.8</td>
<td>68.9</td>
<td>68.9</td>
<td>3.2</td>
</tr>
<tr>
<td></td>
<td>both</td>
<td>52.9</td>
<td>68.6</td>
<td>72.6</td>
<td>73.5</td>
<td>4.5</td>
</tr>
<tr>
<td>ensembled</td>
<td>both</td>
<td>52.9</td>
<td>68.6</td>
<td>74.6</td>
<td><strong>81.9</strong></td>
<td>11.0</td>
</tr>
</tbody>
</table>

**4.2 Publicly Available PLMs**

We experiment with four different-sized PLMs that are publicly available as follows: **Codex** (Chen et al., 2021) The Codex models are descendants of GPT-3 models that can understand and generate code. Their training data contains both natural language and billions of lines of public code from GitHub. We use the version code — davinci — 002 of Codex.

**OPT-175b** (Zhang et al., 2022) Open Pre-trained Transformers (OPT) is a suite of decoder-only pre-trained transformers ranging from 125M to 175B parameters trained on publicly available datasets. We use the version 175 billion parameters of OPT.

**UL2** (Tay et al., 2022) Unified Language Learner (UL2) is 20 billion parameter novel language pre-training paradigm that improves the performance.
Table 3: VQA results on the OK-VQA benchmark comparing to standard baselines. “Wiki” stands for “Wikipedia” and the “Wiki” resource in the last row’s block is brought by the answer selector KAT. “All 4 Frozen PLMs” means that we use all the answer choices generated by GPT-J, UL2, OPT, and Codex. When we have UnifiedQA or KAT as answer selector, we train with 3 random seeds and denote the results as ensemble following (Gui et al., 2022).

<table>
<thead>
<tr>
<th>Method</th>
<th>External Knowledge Source</th>
<th>Answer Selector</th>
<th>Acc(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MUTAN+AN (Ben-Younes et al., 2017)</td>
<td>Wiki</td>
<td>-</td>
<td>27.8</td>
</tr>
<tr>
<td>ConceptBERT (Gardères et al., 2020)</td>
<td>ConceptNet</td>
<td>-</td>
<td>33.7</td>
</tr>
<tr>
<td>KRISP (Marino et al., 2021)</td>
<td>Wiki+ConceptNet</td>
<td>-</td>
<td>38.9</td>
</tr>
<tr>
<td>MAVEx (Wu et al., 2022)</td>
<td>Wiki+ConceptNet+Google Images</td>
<td>-</td>
<td>39.4</td>
</tr>
<tr>
<td>PiCa (Yang et al., 2022)</td>
<td>Frozen GPT-3 Wiki</td>
<td>-</td>
<td>48.0</td>
</tr>
<tr>
<td>KAT (Gui et al., 2022) (ensemble)</td>
<td>Wiki+Frozen GPT-3 Wiki</td>
<td>-</td>
<td>54.4</td>
</tr>
<tr>
<td>ClipCap (Mokady et al., 2021)</td>
<td>-</td>
<td>-</td>
<td>22.8</td>
</tr>
<tr>
<td>RAS0</td>
<td>Frozen GPT-J</td>
<td>ClipCap</td>
<td>29.5</td>
</tr>
<tr>
<td></td>
<td>Frozen UL2</td>
<td>-</td>
<td>33.1</td>
</tr>
<tr>
<td></td>
<td>Frozen OPT</td>
<td>-</td>
<td>31.3</td>
</tr>
<tr>
<td></td>
<td>Frozen Codex</td>
<td>-</td>
<td>35.3</td>
</tr>
<tr>
<td></td>
<td>All 4 Frozen PLMs</td>
<td>-</td>
<td>38.0</td>
</tr>
<tr>
<td>RAS0</td>
<td>Frozen GPT-J</td>
<td>IterPLM</td>
<td>29.6</td>
</tr>
<tr>
<td></td>
<td>Frozen UL2</td>
<td>-</td>
<td>33.8</td>
</tr>
<tr>
<td></td>
<td>Frozen OPT</td>
<td>-</td>
<td>58.5</td>
</tr>
<tr>
<td></td>
<td>Frozen Codex</td>
<td>-</td>
<td>45.7</td>
</tr>
<tr>
<td>RAS0</td>
<td>Frozen GPT-J</td>
<td>UnifiedQA (ensemble)</td>
<td>47.2</td>
</tr>
<tr>
<td></td>
<td>Frozen UL2</td>
<td>-</td>
<td>45.8</td>
</tr>
<tr>
<td></td>
<td>Frozen OPT</td>
<td>-</td>
<td>47.8</td>
</tr>
<tr>
<td></td>
<td>Frozen Codex</td>
<td>-</td>
<td>51.2</td>
</tr>
<tr>
<td></td>
<td>All 4 Frozen PLMs</td>
<td>-</td>
<td>46.5</td>
</tr>
<tr>
<td>RAS0</td>
<td>Wiki+Frozen GPT-J</td>
<td>KAT (ensemble)</td>
<td>50.3</td>
</tr>
<tr>
<td></td>
<td>Wiki+Frozen UL2</td>
<td>-</td>
<td>52.2</td>
</tr>
<tr>
<td></td>
<td>Wiki+Frozen OPT</td>
<td>-</td>
<td>53.0</td>
</tr>
<tr>
<td></td>
<td>Wiki+Frozen Codex</td>
<td>-</td>
<td>58.5</td>
</tr>
<tr>
<td></td>
<td>Wiki+ All 4 Frozen PLMs</td>
<td>-</td>
<td>57.9</td>
</tr>
</tbody>
</table>

4.4 Answer Selection Models

We plug in existing text-generation models as answer selectors and experiment on four methods: KAT (Gui et al., 2022) is a VQA method that uses a sequence-to-sequence model composed of an encoder and a decoder, similar to T5 (Raffel et al., 2020). As far as this paper is written, KAT is known to be the SOTA method on OK-VQA benchmark. ClipCap (Mokady et al., 2021) uses the CLIP (Radford et al., 2021) encoding as a prefix to generate textual captions by employing a simple mapping network over the raw encoding, and then fine-tunes a language model to generate a valid caption. The language model we use here is GPT-2. In this pa-
Table 4: Ablation study investing how different inputs influence the answer selection results using KAT (top) and UnifiedQA (bottom) on OK-VQA in accuracy scores. “Top1” means using Top 1 answer choice, “All” in the first row means using all answer choices, to form the input respectively. “cot” means the CoT rationales. We train with 3 random seeds and denote the average scores as single and majority vote results as ensemble. “All” in the leftmost column represent using combined answer choices from all four PLMs.

<table>
<thead>
<tr>
<th></th>
<th>KAT</th>
<th>GPT-J (single)</th>
<th>GPT-J (ensemble)</th>
<th>UL2 (single)</th>
<th>UL2 (ensemble)</th>
<th>OPT (single)</th>
<th>OPT (ensemble)</th>
<th>Codex (single)</th>
<th>Codex (ensemble)</th>
<th>All (single)</th>
<th>All (ensemble)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Top1</td>
<td></td>
<td>45.9</td>
<td>46.6</td>
<td>50.2</td>
<td>51.1</td>
<td>51.7</td>
<td>52.1</td>
<td>56.2</td>
<td>57.1</td>
<td>56.4</td>
<td>57.0</td>
</tr>
<tr>
<td>All w/o cot</td>
<td></td>
<td>47.8</td>
<td>48.4</td>
<td>50.7</td>
<td>51.5</td>
<td>52.3</td>
<td>52.9</td>
<td>57.1</td>
<td>58.1</td>
<td>56.9</td>
<td>57.6</td>
</tr>
<tr>
<td>All w/ cot</td>
<td></td>
<td>49.6</td>
<td>50.3</td>
<td>51.2</td>
<td>52.2</td>
<td>52.5</td>
<td>53.0</td>
<td>57.5</td>
<td>58.5</td>
<td>57.0</td>
<td>57.9</td>
</tr>
</tbody>
</table>

Table 5: Ablation study on how different inputs influence the answer selection result using IterPLM: iterative prompting using the same PLM, on OK-VQA. All results are in accuracy scores. Both setting use all the answer choices.

<table>
<thead>
<tr>
<th>Type</th>
<th>GPT-J</th>
<th>UL2</th>
<th>OPT</th>
<th>Codex</th>
</tr>
</thead>
<tbody>
<tr>
<td>DG w/o cot</td>
<td>28.5</td>
<td>29.1</td>
<td>31.6</td>
<td>45.6</td>
</tr>
<tr>
<td>DG w/ cot</td>
<td>28.1</td>
<td>32.3</td>
<td>33.5</td>
<td>44.9</td>
</tr>
</tbody>
</table>

Table 6: Ablation study on how different inputs influence the answer selection result using ClipCapVQA (Mokady et al., 2021) on OK-VQA. The first column represents two CLIP checkpoints. “DG” represents direct generation without any answer choices.

<table>
<thead>
<tr>
<th>Type</th>
<th>GPT-J</th>
<th>UL2</th>
<th>OPT</th>
<th>Codex</th>
</tr>
</thead>
<tbody>
<tr>
<td>ViT-L_14 w/o cot</td>
<td>23.5</td>
<td>29.7</td>
<td>30.3</td>
<td>31.3</td>
</tr>
<tr>
<td>ViT-L_14 w/ cot</td>
<td>29.5</td>
<td>33.1</td>
<td>31.3</td>
<td>35.3</td>
</tr>
<tr>
<td>RN50x64 w/o cot</td>
<td>21.6</td>
<td>29.3</td>
<td>30.3</td>
<td>31.4</td>
</tr>
<tr>
<td>RN50x64 w/ cot</td>
<td>29.6</td>
<td>32.6</td>
<td>28.6</td>
<td>34.5</td>
</tr>
</tbody>
</table>

4.5 End-task VQA Results

As illustrated in Table 3, we compare our proposed pipeline against several standard baseline approaches: MUTAN+AN (Ben-Younes et al., 2017), ConceptBERT (Gardères et al., 2020), KRISP (Marino et al., 2021), MAVEx (Wu et al., 2022), PiCA (Yang et al., 2022), and KAT (Gui et al., 2022), on the OK-VQA data test set. RASO outperforms the previous SOTA by an absolute 4% margin, achieving the new SOTA.

Comparing different answer selectors, it is surprising that the two transformer-based text-only models: UnifiedQA and KAT significantly outperform the multi-modal ClipCap model by around 20% on average, even though their sizes (T5 large) are much smaller than that of GPT-2. We believe this phenomenon is because the Clip image embeddings trained using image captions do not have enough granularity to support reasoning over the image, question, and answer choices for answer selection, compared to T5 models. Besides, IterPLM has much worse scores than we imagined. While many papers (Wang et al., 2022) show that iterative prompting should boost the performance, our experiments suggest that asking the PLMs to select between their own output at the highest confidence is not beneficial.
is indeed a very difficult problem for them. In Table 3, we also compare single PLM answer choices with ensembled choices by all four PLMs, with the latter showing lower scores. We believe this is because the answer selectors we experiment on are not good enough, and thus increasing choice numbers turns out to hurt the performance.

4.6 Implementation Details

In the answer choice generation step, we use 16-shot in-context examples on the test data. On the training data, because we have ten gold answers with repetitions, we use 4-shot in-context learning for faster generation. The temperature for PLM generation is set to be 0.001. The generation max token length is set to be 15. All experiments of selection models have been run in 8 NVIDIA V100 Tensor Core GPUs with 32 GiB of memory each, 96 custom Intel Xeon Scalable (Skylake) vCPUs, and 1.8 TB of local NVMe-based SSD storage. The running times for KAT, UnifiedQA and ClipCap are less than 4, 2 and 1 hours, respectively. OPT-175b model is locally set up in 32 NVIDIA V100 Tensor Core GPUs to make inferences. The learning rates for KAT, UnifiedQA and Clipcap are set as 3e-5, 5e-5 and 2e-5, respectively, for all experiments. Optimizer AdamW (Loshchilov and Hutter, 2017) is used for all selection models.

5 Ablation Studies

We perform qualitative and quantitative analysis on the answer selection results to better understand whether the expanded external knowledge coverage benefits the end-task VQA much. As illustrated in Tables 4 to 6, we investigate the impact of various inputs on the answer selection results, with answer choices representing the retrieved knowledge.

CoT Rationale Impact From the experiments results in Tables 4 to 6 where we compare the settings: “w/cot” and “w/o cot”; input with CoT rationales consistently boosts the answer selection performance of KAT, UnifiedQA, and ClipCap. However, this conclusion fails for iterative prompting – adding CoT hurts the performance of IterPLM when we use GPT-J and Codex. We believe this can result from the difference in CoT qualities, and different pre-training methods and data.

Choice Number Impact As shown in Table 4, larger knowledge coverage, represented by using choices from all four PLMs versus a single PLM, can not consistently increase the performance of KAT or UnifiedQA. As we compare the results on Codex choices and that on all PLMs choices, more choices always lead to lower accuracy scores. This is somehow against our instinct, and we believe it is because our answer selectors are not good enough. Digging deeper into the problem, we further compare the difference between using Top1 choices and all choices in KAT as in the top table. Note that the Top1 results here are not the same as the Top1 accuracy in Table 2 because KAT uses Wikipedia knowledge by design so it further expands knowledge coverage. We can see that using all choices is consistently better than using Top 1 choice. However, the improvements are too small (0.4-1.9 %) considering that their knowledge coverages differ by at least 20% as in Table 1, suggesting that KAT, while being the best, is still not the ideal selection model, and motivating future research in this direction.

Multi-modal Selector Impact As demonstrated in Table 6, we experiment with the two versions of CLIP embedding: “ViT-L_14” and “RN50x64” and the difference between direct generation (DG) and answer selection is constantly large – providing answer choices definitely helps ClipCap to generate the correct answer.

Ensemble Impact Our answer choice generation step is indeed ensembling on PLMs results. Previous VQA methods that retrieve from PLMs also conduct ensembling but in a different way (Yang et al., 2022). They usually request the same prompt (see example in Figure 4) multiple times and take the majority-voted answer. This process is called multi-query ensemble, and could boost the GPT-3 performance by about 5%. We argue that our proposed RASO prompting is superior to multi-query ensemble in that we allow more diversity in the output and provide VQA systems more explainability by separating the choice-generation and selection steps, without additional API request cost or longer inference time.

6 Conclusion

In this paper, we propose RASO: a new VQA pipeline following a generate-then-select strategy guided by world knowledge. RASO proposes a new prompting method that largely increases the external knowledge coverage by a margin of more than 20% compared to previous approaches on the OK-VQA benchmark. Our pipeline achieves the new SOTA 58.5% on the end-task performance .
encouraging avenues for future studies.

7 Limitations
While the previous VQA methods that retrieve from PLMs all use GPT-3, we do not experiment with GPT-3 in this paper due to the additional cost. We only focus on applying text-generation models as answer selectors, while classification models could also potentially be good answer selectors. The multi-modal CLIP embedding has already been surpassed by several recent studies (Alayrac et al., 2022; Singh et al., 2022; Lu et al., 2022) and therefore ClipCap cannot represent the performance of multi-modal answer selectors.

8 Ethical Considerations
The authors of this paper acknowledge the significance of responsible NLP in research and development. The objective of this research is to enhance the capabilities of visual question answering models, guided by human values-based world knowledge. We strive to ensure that the model is not only accurate and efficient, but also fair and unbiased. We recognize that the VQA technology may have a substantial impact on society and pledge to be transparent in sharing our findings and progress with relevant users and stakeholders.

Acknowledgments
The authors would like to thank researchers at AWS AI Labs who commented on or otherwise supported throughout the course of this project, including Simeng Han, Donghan Yu, Sijia Wang, and Shuaichen Chang.

References


Xingyu Fu, Ben Zhou, Ishaan Chandra, Carl Vondrick, and Dan Roth. 2022. There’s a time and place for reasoning beyond the image. In Proceedings of the 60th Annual Meeting of the Association for


A Appendix

A.1 CoT prompts

For our CoT generation experiments, we use a pre-designed fixed prompt as partly shown in Figure 6.

A.2 Additional Experiments

We conduct additional experiments for RASO on an augmented successor dataset of OK-VQA: A-OKVQA (*Schwenk et al., 2022*) to prove its effectiveness. Since we do not have the baseline results or any intermediate outputs on A-OKVQA as the paper was written, we only compare with PiCA (*Yang et al., 2022*) with a simpler setting: without using image tagging or chain-of-thought and only using GPT-J. The captions we use are generated using BLIP-2 (*Li et al., 2023*), following the default example in the paper.
Figure 6: The fixed prompt we use to generate chain-of-thought style rationales. We randomly select seven examples in the prompt and show two of them here. We set the temperature as 0.7 and max token as 80 during inference for all PLMs.

<table>
<thead>
<tr>
<th></th>
<th>PiCA</th>
<th>RASO</th>
</tr>
</thead>
<tbody>
<tr>
<td>A-OKVQA</td>
<td>33.2</td>
<td>37.1</td>
</tr>
</tbody>
</table>

Table 7: Additional comparison of RASO versus PiCA on A-OKVQA dataset.
A For every submission:

- A1. Did you describe the limitations of your work? 
  
- A2. Did you discuss any potential risks of your work? 
  
  There are few potential risks of our work since we work on a famous benchmark for academic purpose.

- A3. Do the abstract and introduction summarize the paper’s main claims? 
  
- A4. Have you used AI writing assistants when working on this paper? 
  
  Left blank.

B Did you use or create scientific artifacts?

- B1. Did you cite the creators of artifacts you used? 
  
- B2. Did you discuss the license or terms for use and / or distribution of any artifacts? 
  
  Because the models we use are all publically available. The only model that requires an API to request is Codex by OpenAI, and we follow their Use case policy and content policy. All the models we use are purely for scientific research purpose. The dataset we use is also public.

- B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)? 
  
- B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it? 
  
  The dataset we use: OK-VQA, is famous and used in a lot of papers. It does not include any unique individual privacy information or offensive content.

- B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.? 
  
  Because it’s not very relevant to our paper.

- B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be. 
  
  We report statistics in Section 4.1.

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.
C ✓ Did you run computational experiments?
   
4, 5

✓ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
   
4

✓ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
   
4

✓ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
   
4, 5

✓ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
   
4, 5

D ❌ Did you use human annotators (e.g., crowdworkers) or research with human participants?

Left blank.

☐ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
   
Not applicable. Left blank.

☐ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants’ demographic (e.g., country of residence)?
   
Not applicable. Left blank.

☐ D3. Did you discuss whether and how consent was obtained from people whose data you’re using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
   
Not applicable. Left blank.

☐ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
   
Not applicable. Left blank.

☐ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
   
Not applicable. Left blank.