Machine Intelligence Research
www.mi-research.net

23(1), February 2026, 115-132
DOI: 10.1007/s11633-025-1591-z

Bootstrapping Large Language Models with Outside-knowledge for Knowledge-based Visual Question Answering

Yanze Min[1] Yawei Sun[2] Yin Zhu[3] Jun Zhu[1] Bo Zhang[1]

[1] Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China

[2] State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing 210023, China

[3] Alibaba Group, Alibaba, Hangzhou 310013, China

Abstract: Knowledge-based visual question answering (KB-VQA), requiring external world knowledge beyond the image for reasoning, is more challenging than traditional visual question answering. Recent works have demonstrated the effectiveness of using a large (vision) language model as an implicit knowledge source to acquire the necessary information. However, the knowledge stored in large models (LMs) is often coarse-grained and inaccurate, causing questions requiring finer-grained information to be answered incorrectly. In this work, we propose a variational expectation-maximization (EM) framework that bootstraps the VQA performance of LMs with its own answer. In contrast to former VQA pipelines, we treat the outside knowledge as a latent variable. In the E-step, we approximate the posterior with two components: First, a rough answer, e.g., a general description of the image, which is usually the strength of LMs, and second, a multi-modal neural retriever to retrieve question-specific knowledge from an external knowledge base. In the M-step, the training objective optimizes the ability of the original LMs to generate rough answers as well as refined answers based on the retrieved information. Extensive experiments show that our proposed framework, BootLM, has a strong retrieval ability and achieves state-of-the-art performance on knowledge-based VQA tasks.

Keywords: Multi-modal large language models, visual question answering (VQA), knowledge retrieval, graphical models, machine learning.

Citation: Y. Min, Y. Sun, Y. Zhu, J. Zhu, B. Zhang. Bootstrapping large language models with outside-knowledge for knowledge-based visual question answering. *Machine Intelligence Research*, vol.23, no.1, pp.115–132, 2026. http://doi.org/10.1007/s11633-025-1591-z

1 Introduction

Visual question answering (VQA) is a fertile ground for testing AI models for open-ended vision-language understanding[1]. Knowledge-based visual question answering (KB-VQA)[2, 3] is a more challenging task since, in addition to understanding questions and images, the model is required not only to retrieve the necessary knowledge from an outside knowledge base but also to incorporate the knowledge from its original representation for reasoning. A successful KB-VQA model typically requires strong text and visual recognition as well as the ability to reason with external knowledge[4].

Recently, with the advancement of large (vision) language models[5–8], significant progress has been achieved in KB-VQA tasks. These methods[9–11] resort to pre-trained large models (LMs) as an additional knowledge source for knowledge acquisition. Rich commonsense knowledge has been implicitly encoded in the model parameters, thanks to the enormous training data collection. However, for questions requiring external finer-grained information, e.g., Fig. 1(a), LMs often fail to give the correct answer due to lacking specific domain knowledge. To answer questions such as in Fig. 1(a), the model requires not only commonsense knowledge of animal types (sheep) but also domain-specific knowledge of events (the State Fair of Texas) while noticing iconic yellow tags on sheep's ears. As a result, these existing pipelines (Fig. 1(b)) apply an extra VQA model for either pre-processing (e.g., generating candidate answers) or post-processing (e.g., taking the frozen LM's answer as context) to help the frozen or finetuned LM. However, although LMs are expert at generating continuous descriptive sentences, research shows that they tend to produce hallucinations when lacking sufficient evidence[13], which hinders the KB-VQA performance.

Research Article
Manuscript received on April 28, 2025; accepted on August 11, 2025
Recommended by Associate Editor Hao Sun
Colored figures are available in the online version at https://link.springer.com/journal/11633

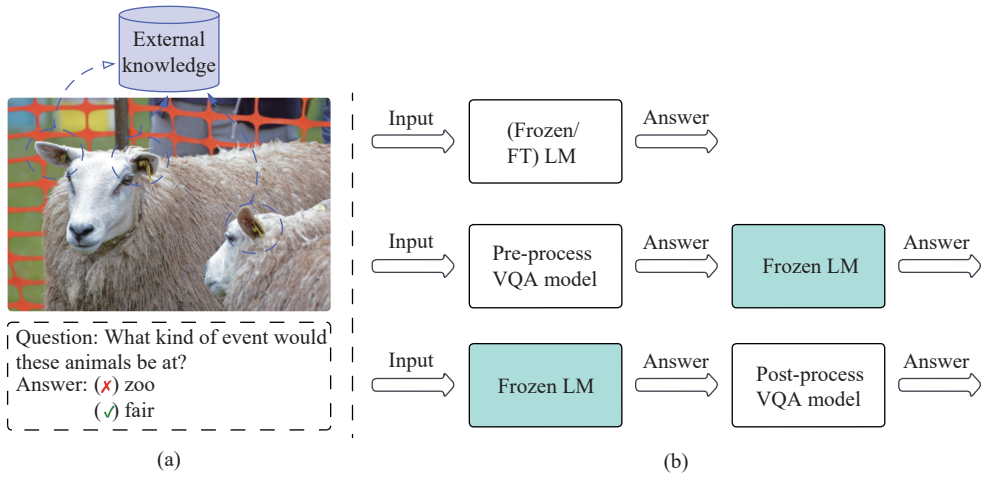

Fig. 1 KB-VQA tasks and typical processing pipelines. (a) A KB-VQA example. (b) Existing pipelines for solving KB-VQA with LMs. Besides directly prompting the LM[11], most works apply an extra VQA model for pre-processing (e.g., generating candidate answers[9]) or post-processing (e.g., taking the frozen LM's answer as context[12]) to help the frozen LM. (Colored figures are available in the online version at https://link.springer.com/journal/11633)

To this end, we present a variational EM framework called BootLM that enhances the KB-VQA performance of LMs via its own answers with the help of external KBs. As shown in Fig. 2(a), instead of asking the final answer directly from LMs, our framework takes advantage of the LM's ability to give rough answers as an intermediate layer, e.g., general descriptions or educated guesses. Compared to traditional methods that infer answers directly (Fig. 2(b)), we introduce two latent variables to the pipeline (Fig. 2(c)): rough answer z and required external knowledge d. We infer these variables in the E-step via an approximated posterior. Specifically, one term of the posterior $p^{(LM)}(z|y,x)$ leverages the LM itself, and the other term $p^{(IR)}(d|z,y)$ is defined by a late-interaction neural retriever. We approximate the M-step likelihood objective with Monte-Carlo samples of the posterior. The objective bootstraps the answer-generation ability of LMs in a cyclic manner.

Intuitively, the retrieved information serves as additional background to the question. The finetuned LM, which is now equipped with much more information about the task at hand, will provide more accurate guesses about questions in the similar field and adjust the retriever's behavior accordingly in a self-enhanced fashion. Compared with existing retrieval augmented generation (RAG) methods[4, 14], our method bootstraps LMs with its

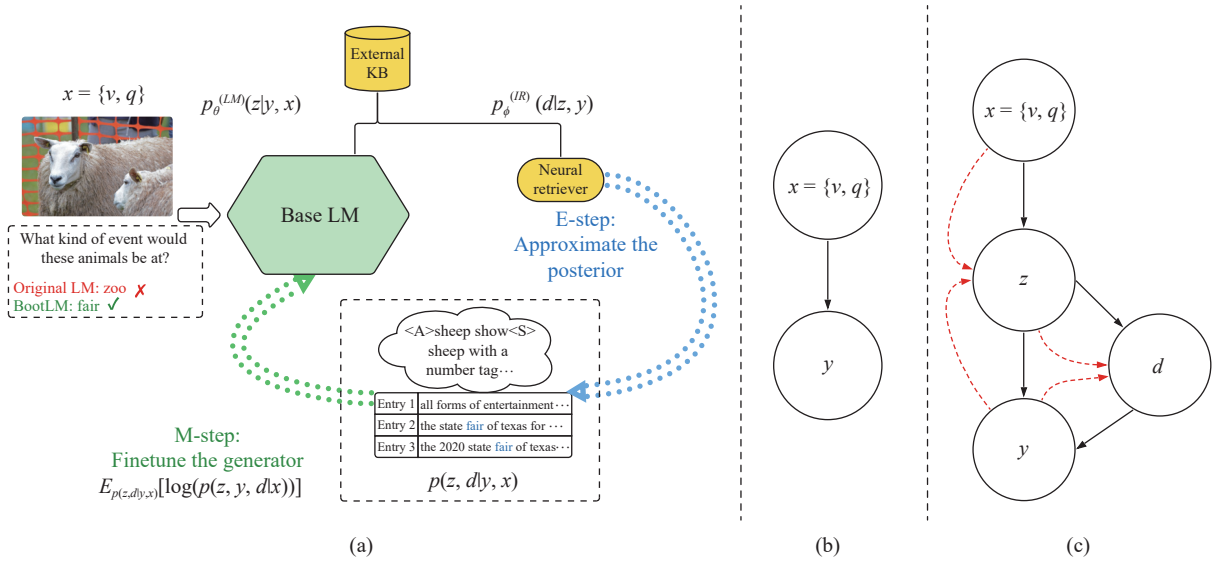

Fig. 2 BootLM framework overview. (a) The general framework of BootLM. BootLM is a general variational EM framework for refining LMs' KB-VQA performance with their own answers. In the E-step, we approximate the posterior with the base LM and a neural retriever. In the M-step, we refine the LM with knowledge from external KBs. (b) Graphical model of traditional VQA models. (c) Graphical model of BootLM. Red arrows correspond to the inference paths. (Colored figures are available in the online version at https://link.springer.com/journal/11633)

own output, so we do not necessarily require additional feature extractors or VQA process models as in Fig. 1(b), though it is straightforward to combine these methods into our framework. Another major advantage of BootLM is that the "rough answer" generator and the final answer generator share the same set of parameters and are refined jointly in the bootstrapping cycle, which unifies the observation behavior and the generation behavior in a consistent manner.

In summary, our contributions are as follows: 1) We formulate a principled EM framework, BootLM, for bootstrapping LMs through its rough answers on KB-VQA tasks. 2) We enhance the LM with additional information retrieval paths in a novel way, combining implicit knowledge from LMs with explicit knowledge from KBs. 3) Extensive experiments show that our proposed framework, BootLM, achieves state-of-the-art (SOTA) retrieval and KB-VQA performance.

2 Related work

VQA. Given an image and a question, the task of VQA is to analyze the visual context and generate answers[1, 15]. Early methods extract image and question features and then fuse them by simple operations (e.g., concatenation) or neural networks (e.g., long short-term memory (LSTM))[1, 16, 17]. Attention-based VQA models[18–21] aim to selectively focus on the most relevant regional image or text features, thus achieving higher performance. Neural module networks (NMNs)[22–24], consisting of a set of shallow neural networks called "modules", are specially designed to handle the compositional structure of questions. To learn and reason about relationships between objects, graph-based approaches[25–27] also use abstract or concrete graph representations of inputs. Recent years have witnessed the rapid development of large pre-trained vision-language models, such as OSCAR[28], CLIP[29], and GLIP[30]. Several works successfully utilize these pre-trained models and then fine-tune on VQA tasks[31–33]. Following these works, we propose a novel VQA training procedure to refine the coarse-grained answers iteratively.

KB-VQA approaches. KB-VQA could access both structured data (e.g., Wikidata) and unstructured data (e.g., Wikipedia)[3, 34–38]. Various multimodal approaches have been investigated for the KB-VQA task. Specifically, KRISP[4] leverages the implicit reasoning of transformer models and integrates symbolic representations from KB. KAT[39] incorporates both implicit and explicit knowledge within an encoder-decoder architecture. VRR[40] is a visual retriever-reader pipeline. TRiG-Ensemble[41] transforms images into plain texts to retrieve knowledge and generate answers. MuKEA[42] uses an explicit triple that correlates visual objects and answers by implicit relations. Prophet[9] prompts LLM with complementary answer heuristics for KB-VQA. Unlike these approaches, our approach solves KB-VQA by bootstrapping LMs.

Multimodal large language model (MLLM). MLLMs have emerged as a research topic with a rich set of open-source MLLMs[6, 43, 44]. When fine-tuned on vision-language instructions, MLLMs achieve strong performance in vision-language tasks. There are numerous well-known MLLMs available online, such as MiniGPT-v2[6], InstructBLIP[43], PaLM-E[45], Flamingo[46], and LLaVA[44]. Several studies propose retrieval-augmented[47] pipeline or prompt engineering[48, 49] to integrate knowledge in a more scalable manner. Specifically, RA-CM3[47] uses an MLLM to refer to relevant texts and images using a retriever from external memory. Q&A Prompts[48] encodes the generated question-answer pairs with a visual-aware prompting module and sends them into MLLMs for VQA. GeReA[49] prompts an MLLM with a question-aware vision to generate descriptions and reasons for these descriptions for KB-VQA. BootLM introduces explicit knowledge sources into LMs, thus enhancing the reasoning capability.

Knowledge retrieval. Knowledge retrieval methods in VQA aim to retrieve top-K relevant documents from external KBs. Among recent retrieval techniques, dense passage retrieval[50–52] methods learn one-dimensional embeddings and compute similarities between questions and documents. ColBERT[53] proposes multi-dimensional embeddings to represent fine-grained relevance. REAVL[54] leverages KBs to assist vision-language pre-training. In particular, RAG[55] has emerged as a pivotal pipeline in enhancing the capabilities of LLMs by enabling access to external knowledge sources. MuRAG[56] is an early work that constructs a database combining retrieval and augmentation to generate better outputs. UDKAG[57] uses a similar architecture but places emphasis more on the recency of knowledge. FLMR[14] leverages multi-dimensional representations to capture fine-grained, cross-modal relevance between text and images more effectively. Following this line of work, RMR[58] enhances reasoning capabilities by introducing in-context learning within the multimodal RAG framework. Meanwhile, RagLLaVA[59] incorporates knowledge-enhanced re-ranking and noise injection during training to improve model robustness. In comparison, our work is the first to formulate a principled probabilistic graphical model framework by introducing extra latent variables. We also propose approximating a factor of the posterior with the LLM itself, allowing the model to refine in a bootstrapping fashion.

3 Methodology

In this section, we first introduce BootLM, a general framework for refining LMs' KB-VQA performance with their own answers. The general framework is illustrated in Fig. 2. To combine the domain knowledge from external KBs with general knowledge encoded in LMs, we for-

mulate the joint distribution of the rough answer z, the retrieved information d, and the refined answer y under the framework of the variational EM algorithm, which is trained alternately between an E-step and an M-step. In the E-step, we derive an efficient approximation of the posterior distribution using parameters of the original LM. We employ a late-interaction retriever to infer the relevant domain knowledge, during which the information preserved by LMs can be effectively distilled into the retriever. In the M-step, we fine-tune the LM based on both the rough answer and the retrieved knowledge. Then, we give detailed descriptions of each module.

3.1 Bootstrapping large models

Given an input $x = \{v, q\}$ with image v and question text q, the goal of VQA is to generate a correct answer y. A straightforward solution is to model $p_\theta(y|x)$ as in Fig. 2 (b); however, the method cannot easily incorporate domain knowledge. The graphical model of the proposed BootLM framework is illustrated in Fig. 2 (c), with two extra random variables z and d, where z denotes the rough answer (e.g., general description or educated guess), and d denotes the external domain knowledge retrieved from KBs. We can derive the conditional joint distribution as follows:

$$p(z, y, d|x) = p(z|x)p(d|z)p(y|d, z). \tag{1}$$

The feed-forward process of BootLM depicted in Fig. 2 (c) is more plausible compared to Fig. 2 (b) when answering complex questions that require external domain knowledge. The process is inspired by the process that humans use to answer difficult questions: We first get a rough answer to the question, then search for relevant external information, and finally propose a carefully considered answer.

For optimizing $p(y|x)$, we follow the EM paradigm and treat z and d as latent variables to be inferred. In the E-step, we aim to infer the posterior distribution $p(z, d|x, y)$. Since the exact inference needs to sum over all possible states of latent variables, we propose to approximate the exact posterior with the following equations:

$$p(z, d|y, x) = p(z|y, x)p(d|z, y) \approx \tag{2}$$

$$p_\theta^{(LM)}(z|y, x)p_\phi^{(IR)}(d|z, y) \tag{3}$$

where $p_\theta^{(LM)}$ denotes the distribution induced by the LM and $p_\phi^{(IR)}$ denotes the retriever distribution. Both modules will be explained in more detail in Sections 3.2 and 3.3. A key feature of our approach is to use the LM itself to model $p(z|y, x)$ rather than introducing a separate module. Since LMs are trained with a large corpus, it is in their ability to generate rough answers directly without external help. Using the same LM to model both $p(z|y, x)$ and $p(y|d, z)$ (i.e., the generation module) can not only simplify the architecture but also update the observation parameters and the generation parameters in a consistent manner.

Note that in this step, we fix the LM's parameter θ to θ_{old} of the last iteration, and the only trainable parameter is ϕ. The primary consideration here is to reduce computational complexity. We leave all the updates for the LM parameter in the M-step, and as a result, only the retriever module needs to be adjusted in each E-step. This is a reasonable simplification since the LM itself is already pre-trained with a large corpus to produce relatively high-quality $p_\theta^{(LM)}(z|y, x)$. As we will show in the experiments, tuning $p_\phi^{(IR)}(d|z, y)$ alone conditioned on the LM's output results in high retrieval performance.

In the M-step, we fix the retriever $p_\phi^{(IR)}$ and update the parameter θ of the LM. We are essentially optimizing the likelihood function $\mathbb{E}_{p(z,d|y,x)}[\log p(z, y, d|x)]$. As the closed-form solution is intractable, again, we seek approximate computation as we estimate the posterior $p(z, d|y, x)$ with samples:

$$\mathbb{E}_{p(z,d|y,x)}[\log p(z, y, d|x)] \approx \tag{4}$$

$$\frac{1}{N} \sum_{\hat{z},\hat{d} \sim p(z,d|y,x)} \log p(\hat{z}, y, \hat{d}|x) = \tag{5}$$

$$\frac{1}{N} \sum_{\hat{z},\hat{d} \sim p(z,d|y,x)} \{\log p_\theta^{(LM)}(\hat{z}|x) + \log p_\theta^{(LM)}(y|\hat{d}, \hat{z})\} + const. \tag{6}$$

The objective in (6) derived above can be easily optimized through gradient-based fine-tuning techniques, where $\hat{z}$s and $\hat{d}$s are sampled from the distribution $p(z, d|y, x)$ according to the formula in (3). The M-step objective consists of two major components: One corresponds to generating the rough answer with only the input x, and the other corresponds to generating the final answer.

Note that although distributions $p_\theta^{(LM)}(z|x)$, $p_\theta^{(LM)}(y|d, z)$ and $p_\theta^{(LM)}(z|y, x)$ share the same LM and hence the same set of parameters, they are different distributions distinguished by the specific prompt used to generate answers. For example, for $p_\theta^{(LM)}(z|x)$, we may use the following prompts: "Summarize this image in a few words.", "use a few words to illustrate what is happening in the picture." or "could you use a few words to describe what you perceive?". For $p_\theta^{(LM)}(z|y, x)$, we additionally provide label information as prompts to the model. For $p_\theta^{(LM)}(y|d, z)$, we use instructions like "based on the information above, answer the following question with

a short answer".

During training, we iteratively perform the E-step and the M-step until convergence. In this way, we combine the implicit knowledge encoded in LMs with the explicit knowledge provided by KBs, pushing the LM to produce outputs from coarse-grained to fine-grained answers. During the evaluation, we use ancestral sampling in (1) to generate answers: We sample variable z with the LM, retrieve variable d with the neural retriever, and finally yield the answer y through the generation module $p(y|d,z)$.

3.2 Information retrieval module $p_\phi^{(IR)}$

In this section, we introduce the information retrieval module $p_\phi^{(IR)}$. We use the same retriever for modeling $p_\phi^{(IR)}(d|z,y)$ and $p_\phi^{(IR)}(d|z)$, distinguished by whether to incorporate the ground truth y as the input.

As illustrated in Fig. 3, the retrieval module is equipped with an encoder model $Enc(\cdot)$ and a late interaction neural retriever[14, 53, 60] $R(\cdot)$. For an overview, the encoder takes as input a query sequence $\mathbb{Q}$, which is constructed with various prompts according to different settings. It outputs the query embedding $Enc(\mathbb{Q})$. The corpus embedding $\mathbb{C}$ consists of embeddings of every document in the KB. The neural retriever computes the relevant score $R(Enc(\mathbb{Q}),\mathbb{C})$. The distribution $p_\phi^{(IR)}$ is defined based on the relevant score. The module is fine-tuned at each iteration following the popular pseudo-label heuristic that a document is considered pseudo-relevant if it contains any human-annotated answers.

To be more concise, we take the computation of $p_\phi^{(IR)}(d|z,y)$ as a concrete example. The module starts with constructing the query sequence $\mathbb{Q}$. In this scenario, we first prompt the LM to get

$$z = \{Summary(x), Description(x), DirectAnswer(x)\}$$

where $Summary(x)$, $Description(x)$ and $DirectAnswer(x)$ are outputs of different prompts. For example, we prompt the LM with "A short image caption:" for $Summary(x)$, with "Please describe the scene as detailedly as possible" for $Description(x)$ and "Answer the question with a short answer:" for $DirectAnswer(x)$. We add the answers into $\mathbb{Q}$, and the ground truth answer y is then concatenated to $\mathbb{Q}$.

The neural encoder $Enc(\mathbb{Q})$ yields the token-level embedding of shape $[l_s, h]$ where l_s is the sequence length and h is the embedding size. Similarly, for corpus embedding $\mathbb{C}$, the shape is $[l_e, h]$ where l_e is the number of passage tokens. We compute the relevant score of $Enc(\mathbb{Q})$ and $\mathbb{C}$ as follows:

$$R(Enc(\mathbb{Q}),\mathbb{C}) = \sum_{i=1}^{l_s} \max_{j=1}^{l_e} Enc(\mathbb{Q})_i \mathbb{C}_j^{\mathrm{T}}. \tag{7}$$

The late interaction mechanism aligns each query token with the most contextually relevant passage token, quantifies these matches and combines the partial scores across the query[60]. Compared with single-vector representations such as DPR[50], late interaction retrievers produce richer multi-vector representations, which are more

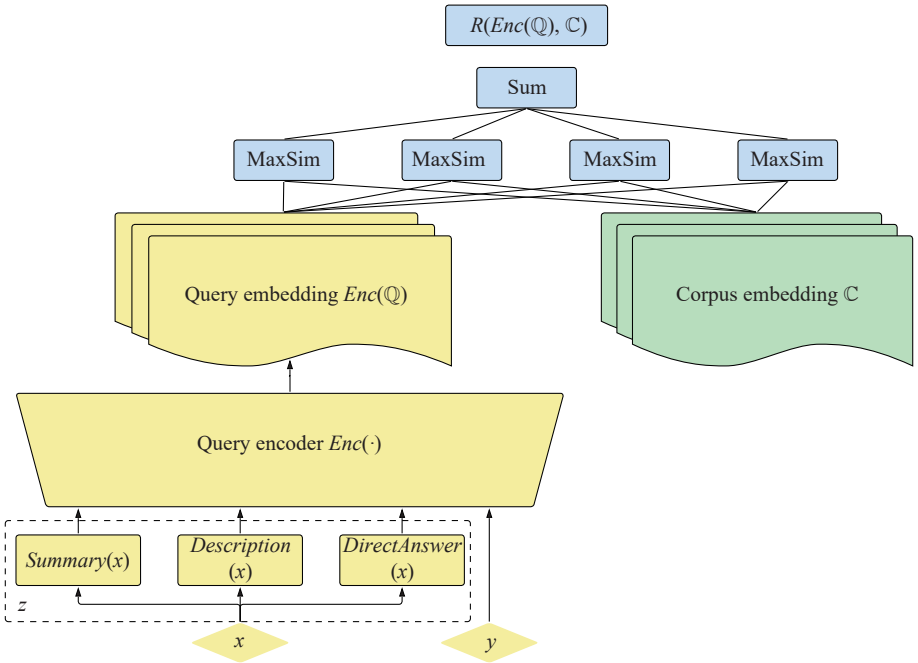

Fig. 3 Information retrieval module. It consists of an encoder model $Enc(\cdot)$ and a late interaction neural retriever $R(\cdot)$. (Colored figures are available in the online version at https://link.springer.com/journal/11633)

suitable to match the competent text generation ability of LMs. Note that the specific form of the query encoding $Enc(\mathbb{Q})$ can be very flexible depending on the base LM. For example, when built upon a base LM which can effectively handle visual grounding tasks, such as MiniGPT-v2, we can also ask the LM to produce bounding boxes of interesting regions, send the image patches into vision encoders, and add the extra output vision features to $Enc(\mathbb{Q})$.

We define the distribution $p_\phi^{(IR)}(d|z,y)$ as a softmax of the relevant score:

$$p_\phi^{(IR)}(d|z,y) = \frac{\mathrm{e}^{R(Enc(\mathbb{Q}(z,y)),d)}}{\sum_i \mathrm{e}^{R(Enc(\mathbb{Q}(z,y)),d_i)}} \tag{8}$$

where ϕ denotes all the parameters in the retriever module, and $Enc(\mathbb{Q})$ denotes the query constructed from the rough answer z and the ground truth y. Following [50], we adopt the pseudo-label heuristic and train ϕ with in-batch negative sampling during each E-step:

$$\mathcal{L}_\phi = -\sum_{\hat{x},\hat{y}\sim p_{data}(x,y),\hat{z}\sim p^{(LM)}(z|\hat{y},\hat{x}),\hat{d}\sim U_{PL}(d|\hat{y})} \log p_\phi^{(IR)}(\hat{d}|\hat{z}) \tag{9}$$

where $p_{data}(x,y)$ is the data distribution and $U_{PL}(d|\hat{y})$ is a uniform distribution defined on the set of all pseudo-labels of $\hat{y}$.

3.3 Generation module $p_\theta^{(LM)}$

We utilize one LM for modeling all the distributions regarding text generation: $p^{(LM)}(z|x)$, $p^{(LM)}(y|d,z)$ and $p^{(LM)}(z|y,x)$. In principle, our framework can take any LMs. As a representative example, we adopt the MiniGPT-v2[6] in this work as the foundation LM since it is designed as a unified interface for vision-language multi-task learning. MiniGPT-v2 comprises a visual ViT backbone, a linear projection layer, and LLaMA-2 language model[7]. The model achieves strong performance on a wide range of VQA tasks.

We fine-tune the LM during training with the approximated M-step likelihood in (6). Since the two terms inside the summation are all log-densities, we can rewrite (6) into a traditional cross-entropy loss for text generation:

$$\mathcal{L}_\theta = -\sum_{(x^*,y^*)\in\mathbb{D}} \log p_\theta^{(\mathrm{LM})}(y^*|x^*) \tag{10}$$

where $\mathbb{D}$ is a composite training dataset consisting of both $(x,\hat{z})$ pairs and $([\hat{d},\hat{z}],y)$ pairs. $[\hat{d},\hat{z}]$ denotes the prompt we constructed from $\hat{d}$ and $\hat{z}$. Specifically, in this work, we construct the following template with $[\hat{d},\hat{z}]$ to prompt MiniGPT-v2:

$\langle Img\rangle$ [*Image*] $\langle /Img\rangle$ [*Task Identifier*] *Doc* $[\hat{d}]$ $\langle Instruction\rangle$ $\langle /Img\rangle$ [*Based on the above information and the given image, respond to this question with a short answer:*] $\langle$Q$\rangle$ [*Question*] $\langle /Q\rangle$ $\langle$*/Instruction*$\rangle$.

3.4 Pseudocode of the training procedure

As a summary of this section, we include the pseudo-code outlining the full BootLM training procedure in algorithm 1.

Algorithm 1. Training procedure for the BootLM framework

Require: Parameters θ of the base LM $p_\theta^{(LM)}$; parameters ϕ of the retrieval module (defined in Section 3.2) $p_\phi^{(IR)}$; VQA training data with query $x=\{v,q\}$ and answer y; the corpus embedding $\mathbb{C}$

Output: Updated parameters θ and ϕ

```
Initialize θ, φ
While not converged do
  repeat                                              ▷ E-step
    Sample {x̂, ŷ} from the training data
    Generate ẑ from p_θ^(LM)(z|ŷ, x̂) with proper prompts
    Uniformly sample d̂ ∼ U_PL(d|ŷ)
    Compute L_φ using (9)
    Update φ according to ∇L_φ with gradient based methods
  until T_1 iterations
  repeat                                              ▷ M-step
    Sample {x̂, ŷ} from the training data
    Generate ẑ from p_θ^(LM)(z|ŷ, x̂) with proper prompts
    Generate d̂ from p_φ^(IR)(d|ẑ, ŷ) using (8)
    Construct 𝔻 consisting of both (x̂, ẑ) pairs and ([d̂, ẑ], ŷ) pairs
    Compute L_θ using (10)
    Update θ according to ∇L_θ with gradient based methods
  until T_2 iterations
end while
```

4 Experiments

In this section, we evaluate the performance of our proposed BootLM with both the base LM and other state-of-the-art KB-VQA methods. We compare multiple strong baselines with or without external KBs and with or without LMs. We use the widely adopted VQA score as our accuracy metric for quantitative evaluation. We also further evaluate our retrieval performance with KB-VQA methods incorporating retrieval modules such as [14, 40, 52]. For information retrieval experiments, as the ground-truth documents for each query are absent, we use pseudo relevance recall (PRRecall) as the metric following [40]. We also conduct qualitative experiments by showcasing failure cases of the base LM that our

BootLM successfully corrects.

4.1 Datasets

We focus mainly on the OK-VQA[2] dataset, one of the largest KB-VQA datasets. It consists of 14 031 images and 14 055 questions divided into a training set of 9 009 and a test set of 5 046 questions. All questions are manually filtered to ensure that either commonsense or domain-specific knowledge is required to answer the questions. Each data sample is annotated with ten open-ended answers. The task is to generate open-ended answers. For similar reasons as [14], we do not use the A-OKVQA dataset, as it focuses heavily on spatial relationship and visual grounding rather than knowledge reasoning. We use the Google Search Corpus[40] as our external KB, a general and easy-to-use knowledge corpus for the OK-VQA benchmark.

To further demonstrate the generalization capability, we also conduct experiments on another KB-VQA task, the fact-based VQA (FVQA)[3] dataset. The dataset consists of commonsense factoid VQA questions, such as "[Q] Which furniture in this image can I lie on? [A] Sofa". The dataset provides a KB consisting of common sense knowledge triplets such as "(Used For, Sofa, Lie on)", which are extracted from ConceptNet[61], Webchild[62] and DBpedia[63]. Overall, the dataset contains 2 190 images and 5 826 questions.

4.2 Baselines

We compare our model with several strong KB-VQA baselines. We divide baseline methods into two groups depending on whether the method incorporates explicit KB retrieval. 1) Among methods without explicit KBs, T5-large[64] and BLIP2[65] achieve SOTA performance on a wide range of natural language processing (NLP) tasks; PICa[11], Prophet[9], and PromptCap[10] achieve SOTA on OK-VQA by prompting GPT-3[8] with captions; PALI[66], Flamingo[46] and PaLM-E[45] are relatively larger models with over 10 B parameters. 2) Among methods with explicit KBs, ConceptBERT[67], KRISP[4], VRR[40], and MAVEx[37] retrieve information from various KBs with relatively lightweight models; KAT-T5[39], TRiG-Ensemble[41], KAT-Ensemble[39], REVIVE[12], RA-VQA[52] and RAVQA-v2[14] are built upon pre-trained large model checkpoints. For retriever baselines, we compare our method with retrieval modules of competitive RAG models: VRR[40], RA-VQA[52], DPR[50] and FLMR (RA-VQA-v2)[14].

4.3 Evaluation metrics

We evaluate the KB-VQA accuracy by the VQA score[2]:

$$VQAScore(y) = \min\{1, \#(y, GT)/3\}$$

where y is the output of the model and $\#(y, GT)$ denotes the number of exact matches of y in the ground truth GT. VQA score is a widely used metric for open-ended generation as it assigns non-zero scores to less popular answers among human responses. For the FVQA dataset, as only one ground truth answer is available, we evaluate the result using top-1 exact matching for each category.

We evaluate the retrieval module by PRRecall@K[40]. As there is no annotation for the ground truth retrieved document, pseudo-relevance is defined if a document contains any human-annotated answers. PRRecall@K measures whether the retrieved K documents contain at least one pseudo-relevant document:

$$PRRecall@K = \min\{1, \#(K, GT)\}$$

where GT denotes the human-annotated answers of the VQA dataset and $\#(K, GT)$ denotes that how many of the retrieved K documents contain at least one answer in GT.

4.4 Implementation details

We initialize the retrieval module with ColBERTv2[53]. We train the module with a learning rate of 1×10^{-5} and a batch size of 30. The embedding size is set to 128. Instead of directly sampling from the distribution $p(d|z, y)$, we choose 5 samples with the highest density each time. Following [14], in addition to text input, we query the LM at most 10 ROIs if possible (e.g., bounding boxes using MiniGPT-v2's grounding tag). We encode the visual feature within ROIs by vision encoders of VinVL[32]. We also add optical character recognition (OCR) texts as an extra retrieval input.

We use MiniGPT-v2[6] as our foundation LM and initialize the model with the official checkpoint stage 2, which focuses on fine-grained datasets to train the model. We choose MiniGPT-v2 as our base LM not for its effectiveness on KB-VQA, but rather for its design as a unified interface for vision-language multi-task learning, which is supposed to generate good quality "rough answers" utilized by the BootLM framework. Following the original model, the visual backbone remains frozen when fine-tuning the model. We adopt the Llama-2-7B-Chat model. We set the rank $r = 64$ and $\alpha = 16$ for low-rank adaptation (LoRA)[68]. We set the image resolution to 448×448 pixels and the batch size as 10. During the fine-tuning, our configuration includes parameters for the learning rate schedule, initialized learning rate (1×10^{-5}), minimum learning rate (1×10^{-6}), warmup learning rate (1×10^{-6}), and weight decay (0.05). All experiments are conducted on 4 Nvidia A100 GPUs.

4.5 Main results

4.5.1 VQA performance

The results on OK-VQA are summarized in Table 1.

Table 1 Results on OK-VQA. The base models column denotes the LM used by the model, with T5-large(0.77 B) being the boundary. The KBs column denotes both the implicit knowledge source (e.g., pre-trained LMs) and the explicit knowledge source (e.g., external KBs). GI is the abbreviation of Google Images. The best performance is underlined. Our result is in **bold**.

Methods	Base models	KBs	Accuracy
Methods w/o explicit KBs			
T5-large (fine-tuned)	T5-large	T5-large	47.52
BLIP 2 (fine-tuned)	BLIP2	BLIP2	55.44
PICa	GPT-3	GPT-3	48.00
Prophet	GPT-3	GPT-3	61.11
PromptCap	GPT-3	GPT-3	60.40
PALI (17 B)	PALI (17 B)	PALI	64.50
Flamingo (80 B)	Flamingo (80 B)	Flamingo	57.80
PaLM-E (562 B)	PaLM-E (562 B)	PALM-E	<u>66.10</u>
Methods with explicit KBs			
ConceptBERT		ConceptNet	33.66
KRISP		ConceptNet, Wikipedia	38.35
VRR		GoogleSearch	45.08
MAVEx		ConceptNet, Wikipedia, GI	39.40
KAT-T5	T5-large	T5-large, Wikipedia	44.25
TRiG-Ensemble	T5-large	T5-large, Wikipedia	50.50
KAT-Ensemble	GPT-3, T5-large	GPT-3, T5-large, Wikipedia	54.41
REVIVE	GPT-3	GPT-3, Wikipedia	58.00
RA-VQA	T5-large	T5-large, GoogleSearch	54.48
RA-VQA-v2	BLIP 2	BLIP 2, GoogleSearch	62.08
Ours			
MiniGPT-v2 + BootLM	MiniGPT-v2	MiniGPT-v2, GoogleSearch	**62.00**
MiniGPT-v2 (*w/o* BootLM)	MiniGPT-v2	MiniGPT-v2	57.82

The best score to date is 66.10 from PaLM-E(562 B). Our method achieves a comparable performance of 62.00 VQA score with a much smaller base LM (less than 10 B).

For an overview, we first compare our framework with the base model without BootLM. We fine-tune MiniGPT-v2 on the OK-VQA dataset directly, yielding a score of 57.82, which is similar to the performance (56.9) reported in the original MiniGPT-v2 paper. The performance of MiniGPT-v2 without BootLM is worse than many methods shown in Table 1, including methods with explicit KBs (e.g., REVIVE, RAVQA) and methods without explicit KBs (e.g., Prophet, PromptCap), demonstrating that MiniGPTv2 alone is not powerful enough on the KB-VQA task. BootLM framework significantly improves the KB-VQA performance of the base LM to 62.00, with a relative increase of 7.23%.

We also notice that when equipped with BootLM, MiniGPT-v2′s performance exceeds almost all methods without explicit KBs except PALI (17 B) and PaLM-E (562 B). Many of these methods leverage a base LM multiple times larger than ours, e.g., GPT-3 (175 B) and Flamingo (80 B), but yield inferior results. This phenomenon suggests the necessity of incorporating explicit domain knowledge when conducting KB-VQA tasks.

Among methods with explicit KBs, our method and RA-VQA-v2 outperform all other methods by a significant margin. RA-VQA-v2 performs slightly better than ours. Note that as a standard RAG framework, RA-VQA-v2 utilizes multiple powerful third-party models for feature extraction and query construction, while BootLM enhances itself in a bootstrapping fashion (more investigation can be found in Section 4.5.2). Besides, it is observed that a fine-tuned MiniGPT-v2 on its own achieves a higher score than most methods with a smaller base model (VRR, MAVEx, etc.) even though they incorporate external KBs. This emphasizes that the commonsense knowledge encoded by LMs is also essential. BootLM proposes a principled framework to combine the two knowledge sources.

The accuracy of each question type during EM itera-

Table 2 BootLM's results for each knowledge category. The numbers are in VQA score. The abbreviations in the first row represent the names of each category in the dataset.

#EM iter.	V.	B.	O.	Sp.	C.	G.	Pe.	Pl.	Sc.	W.	Oth.	Avg.
0	52.7	53.9	58.6	59.2	60.9	58.5	56.1	61.4	52.8	65.1	56.5	58.0
1	54.0	52.2	58.8	63.6	62.0	58.1	57.4	61.5	53.9	65.1	60.6	59.5
2	53.1	57.1	58.7	65.3	60.6	59.0	59.3	63.3	55.2	66.2	62.2	60.2
3	56.6	65.2	62.4	65.0	63.4	61.5	63.5	61.5	57.3	64.8	63.2	62.0
4	56.3	66.1	63.0	64.7	61.9	61.5	60.2	61.4	57.7	65.0	63.5	61.9

tions is shown in Table 2. Compared with the standard VQA training procedure, our method is a general framework that can trade off between model expressiveness and training time by bootstrapping from the model's own rough answers iteratively. Thus, standard VQA training can be conceptually regarded as a special case of our work with only one iteration (the "rough answer" generator won't receive updates). As shown in Table 2, more iterations typically lead to better results. In our experiments, BootLM converges within 3 to 4 EM iterations. Initializing the retriever module with ColBERTv2 and indexing (a pre-processing step for fast retrieval) require around 2 hours. This is a one-time cost. In each E-step, fine-tuning the retriever module requires around 2 GPU hours (2k steps). In each M-step, fine-tuning MiniGPT-v2 with LoRA requires around 4 GPU hours (5k steps).

4.5.2 Retrieval performance

We analyze the retrieval performance of BootLM in Table 3. VRR, RA-VQA, and DPR encode queries and documents into single-vector representations. As a result, the retrieval performance is greatly hindered as the highest PRRecall@5 and PRRecall@10 are only 83.43 and 90.31 each, yielded by the DPR model with both text and image features. In contrast, the late interaction mechanism used by FLMR (the retrieval method of RA-VQA-v2) and BootLM, where queries and documents are encoded into multi-vector representations at a finer granularity, significantly boosts the PRRecall score. At convergence, BootLM, which relies only on its direct answers with additional OCR texts, has already achieved 85.21 PRRecall@5, which is higher than the VRR, RA-VQA, and DPR baselines. Constructing the full rough answer z with direct answers, summaries, and descriptions significantly improves the metric to 89.06. Adding vision features from ROIs proposed by the LM further improves the PRRecall@5 and PRRecall@10 to 89.12 and 93.64 each, though the improvement is not as significant as the previous row (a relative increase of 0.07% & 0.62% VS. 4.52% & 1.24% for PRRecall@5 & PRRecall@10).

The retriever module demonstrates SOTA retrieval capabilities, though it is not the best-performing model. We would like to emphasize that RA-VQA-v2 (FLMR) combines the strength of multiple SOTA third-party models to construct queries, especially fine-grained local visual features. In comparison, BootLM bootstraps itself with mostly its own output. Though LMs are general

Table 3 PRRecall@K on Google Search Corpus. FLMR is the retrieval module of RA-VQA-v2. The query column denotes the main input for each retrieval module. Caption denotes the output of a captioning model. ROI text/vision denotes the patch caption/visual feature of a detection model. LM ROI denotes the visual feature from ROIs proposed by the base LM. The best result is underlined. The best setting of our method is in **bold**.

Retrieval methods	Query	PRR@5	PRR@10
VRR	Question, caption, image feature	80.40	88.55
RA-VQA-FrDPR	Question, caption, OCR, ROI text	81.25	88.51
RA-VQA	Question, caption, OCR, ROI text	82.84	89.00
DPR	Caption, OCR, ROI text	83.08	89.77
DPR	Caption, OCR, ROI text, image feature	83.43	90.31
DPR	Caption, OCR, ROI text & vision, image feature	82.90	89.95
FLMR	Caption, OCR, ROI text	85.99	92.79
FLMR	Caption, OCR, ROI text, image feature	87.02	92.69
FLMR	Caption, OCR, ROI text & vision, image feature	<u>89.32</u>	<u>94.02</u>
BootLM	OCR, answer	85.21	91.92
BootLM	OCR, answer, summary, description	89.06	93.06
BootLM	OCR, answer, summary, description, LM ROI	**89.12**	**93.64**

models for most vision language tasks, it is unlikely for one single model to achieve optimal performance on most tasks, so the gap is understandable. Nonetheless, the margin is not obvious. Since our focus is bootstrapping LMs for KB-VQA tasks, the retrieval module gives a good enough approximation as a factor of the posterior in (3).

4.5.3 Ablation studies

In Table 4, we conduct ablation experiments to analyze how various retrieval configurations, training procedures and base models affect the final model performance. The data in the upper part of Table 4 demonstrates that as long as the appropriate retrieval query is used, the model is not very sensitive to the specific form of retrieval configuration, though adding more informative queries indeed enhances the performance. This observation is also supported by Table 3. We also summarize the performance with three different training procedures, including fine-tuning only on OK-VQA, finetuning on OK-VQA with KB, and Full BootLM in Table 4. Again, full BootLM training procedure (the last row) achieves the best performance since the framework enables more informative "rough answers" as well as better knowledge incorporation ability through every EM update.

Table 4 Ablation study on different retrieval configurations and training procedures

Retrieval configuration	VQA score
None	57.8
OCR, answer	60.2
OCR, answer, summary, description	61.1
OCR, answer, summary, description, LM ROI	62.0
Training procedure	
Only on OK-VQA	57.9
Training set with KB	59.5
Full BootLM	62.0

4.5.4 Performance on FVQA

We conduct experiments on the FVQA dataset to demonstrate the framework's generalization ability. In these experiments, the KB is constructed from FVQA's fact surface collection. We compare the model with three baselines: The first two are the best-performing models reported in [3], and the third is the LM on the dataset with KB. As shown in Table 5, BootLM outperforms the baselines, demonstrating that it generalizes well to other datasets.

4.6 Qualitative Results

We showcase some representative examples in Fig. 4 to illustrate how our approach enhances KB-VQA. More

Table 5 Results on FVQA. The numbers in brackets denote results in each base categories. The best performance is underlined.

Methods	Top-1 accuracy
FVQA (top-1-QQmapping)	52.6
FVQA (top-3-QQmapping)	56.9
MiniGPT-v2 (*w/o* BootLM)	52.3 (obj. 56.6, scn. 11.6, act. 7.7)
BootLM (Ours)	63.9 (obj. 68.9, scn. 18.7, act. 7.7)

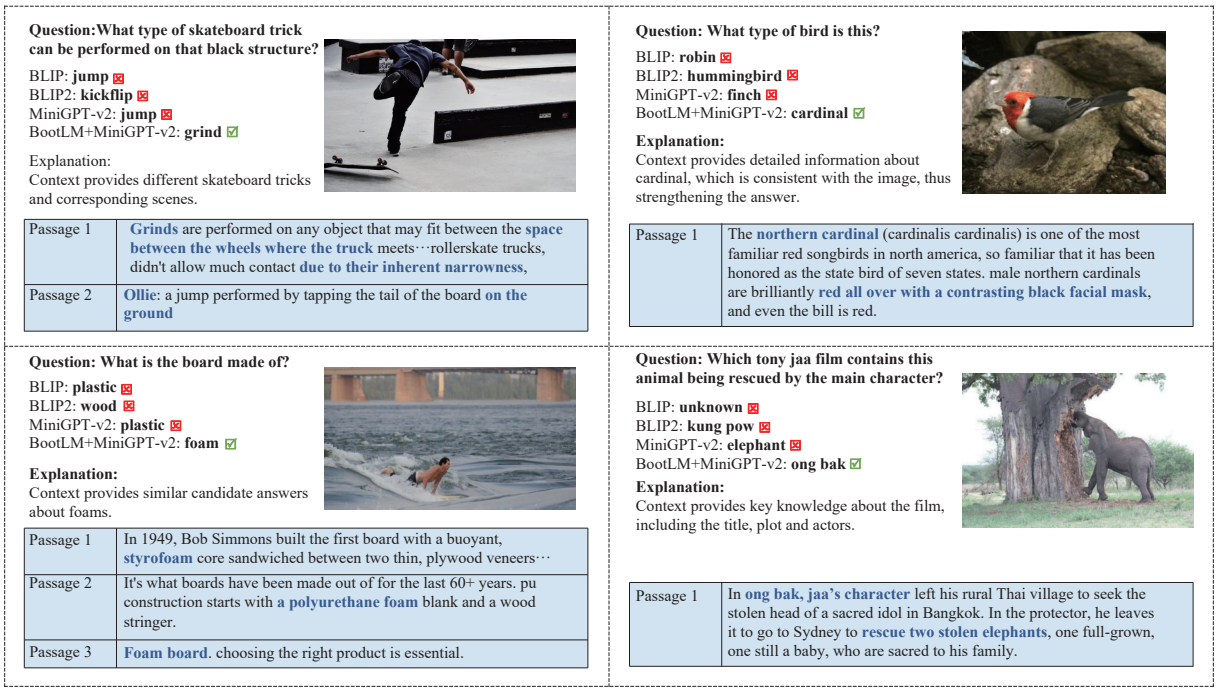

Fig. 4 Qualitative results. We demonstrate the results of BootLM compared to various baselines. (Colored figures are available in the online version at https://link.springer.com/journal/11633)

qualitative examples can be found in Appendix C. In the first example, accurate definitions of candidates are retrieved, helping the model distinguish them. In the second example, the passage describes characteristics consistent with the image, leading to the correct identification. The retrieved context provides similar candidates in the third example, strengthening the correct answer. The final example depicts a challenging case without domain-specific knowledge.

4.7 Error analysis

We randomly sample 50 failure cases of our model and classify errors into 5 categories as follows:

Image detection (16%): Some questions mainly depend on the ability of the vision model, such as identifying species, shapes, and colors. For instance, the question "what shade of blue is the racket?" requires the model to identify the color name accurately.

Logical reasoning (28%): Errors in this category are primarily caused by the lack of reasoning capability of the LM. In most cases, the retrieved context provides sufficient domain knowledge, but the language model fails to organize them into a coherent reasoning chain. For example, the retriever recalls "Victorian design is widely viewed as having indulged in a grand excess of ornament." However, the language model does not relate this information to the description of "a large, ornate pipe organ in a church".

Language model hallucination (6%): The language model generates consistent but unreasonable responses. For example, the language model predicts that a natural disaster of flood, instead of a fire, might occur in a lush green forest.

Knowledge retrieval (8%): Errors in this category occur when the model fails to retrieve relevant knowledge. For example, when asked "when did people start using these (flowers in a vase) as decorations", the retriever should provide relevant folk knowledge, but it may fail to do so.

Others (42%): Other errors are due to small differences between predictions and the ground truth answers for evaluation, such as typos, alternative expressions, or extra descriptions. For instance, the prediction "giraffa camelopardalis", a scientific name, differed from the ground truth "giraffe".

For interested readers, we present more qualitative results in Appendix C, including failure cases. In Appendix D, we discuss the limitations of BootLM in more detail.

5 Conclusions

This work presents a general framework, BootLM, for bootstrapping LMs through their own rough answers on KB-VQA tasks. Through the variational EM framework, we combine the implicit knowledge in LM with explicit knowledge from KBs in a novel way. The effectiveness of BootLM has been demonstrated through both quantitative and qualitative experiments. In future work, we plan to further compensate for the shortcomings of LM′s multi-hop and logical reasoning abilities through neural-symbolic methods. We will also consider more plausible heuristics and more effective ways to estimate the posterior.

Declarations of conflict of interest

Jun Zhu and Bo Zhang are editorial board members of *Machine Intelligence Research* and were not involved in the editorial review, or the decision to publish this article. All authors declare that there are no other competing interests. The authors declared that they have no conflicts of interest to this work.

Appendix A. Detailed derivation for BootLM

Let $x = \{v, q\}$ denote the VQA input. A straightforward pipeline is to model the answer generation distribution $p(y|x)$ directly, where y is the model output. In KB-VQA tasks, external knowledge is often required for successfully solving the question. However, whether pieces of information d in KBs are useful remains unknown. Inspired by humans′ process to answer complex questions, we add another random variable z to the pipeline, representing the "rough answer" that the model yields without incorporating specific domain knowledge. The joint distribution is defined as

$$p(z, y, d|x) = p(z|x)p(d|z)p(y|d, z). \tag{11}$$

We only observe (x, y)s in data, so directly conducting MLE is not an option. Based on the variational EM framework, we separate the optimization into two cyclic steps: An E-step for estimating the posterior of latent variables and an M-step for optimizing the expected likelihood. Specifically in our model, z and d are latent so the E-step is to compute

$$p(z, d|y, x) = p(z|y, x)p(d|z, y). \tag{12}$$

Since the closed-form solution for $p(z, d|y, x)$ is intractable, the variational EM framework seeks an approximated solution q by optimizing $KL(q(z, d)||p(z, d|y, x))$. The design of the variational form of q is rather flexible. In our work, we factorize q into two terms:

$$q(z, d) = p_\theta^{(LM)}(z|y, x)p_\phi^{(IR)}(d|z, y) \tag{13}$$

where $p_\theta^{(LM)}(z|y,x)$ is parameterized by the base LM's θ, and $p_\phi^{(IR)}(d|z,y)$ denotes a retrieval neural network. A feature of BootLM is to utilize $p^{(LM)}$ as a factor in $q(z,d)$ instead of training a separate generation model. As optimizing the KL divergence itself is not a trivial problem, this decomposition leaves the tunable part to the retrieval neural network, which can be efficiently optimized with the widely adopted pseudo-label heuristic.

In the M-step, the expected likelihood function $E_{q(z,d)}\log(p(z,y,d|x))$ is intractable. We estimate the expectation with samples:

$$\begin{aligned}&\mathbb{E}_{q(z,d)}[\log p(z,y,d|x)]\approx\\&\frac{1}{N}\sum_{\hat{z},\hat{d}\sim q(z,d)}\log p(\hat{z},y,\hat{d}|x)=\\&\frac{1}{N}\sum_{\hat{z},\hat{d}\sim q(z,d)}\{\log p_\theta^{(LM)}(\hat{z}|x)+\log p_\theta^{(LM)}(y|\hat{d},\hat{z})\}+const\end{aligned}\tag{14}$$

where in (14), we absorb the term associated with $p(d|z)$ into $const$. In its most strict form, the EM algorithm requires modeling $p(d|z)$ with a separate module without the variational parameter ϕ. However, in our case, the final goal is modeling $p(y|x)$ instead of $p(z,y,d|x)$, so we use the same retriever for both $p(d|z)$ and $p(d|z,y)$.

Appendix B. More experiment details

B.1 Prompt construction

For

$$z=\{Summary(x),Description(x),DirectAnswer(x)\},$$

we prompt the LM with "a short image caption:" for $Summary(x)$, with "please describe the scene in detail" for $Description(x)$ and "answer the question with a short answer:" for $DirectAnswer(x)$.

For the distribution $p^{(LM)}(y|d,z)$, we construct the following template to prompt MiniGPT-v2:

⟨*Img*⟩ [*Image*] ⟨*/Img*⟩ ⟨*Task Identifier*⟩ ⟨*Doc*⟩ [$\hat{d}$] ⟨*/Doc*⟩ ⟨*Instruction*⟩ [*Based on the above information and the given image, respond to this question with a short answer:*] ⟨*Q*⟩ [*Question*] ⟨*/Q*⟩ ⟨*/Instruction*⟩.

As a concrete example, consider the question shown in Fig. B1. We use the following prompt:

⟨*Img*⟩ [*Image*] ⟨*/Img*⟩ [*vqa*] ⟨*Doc*⟩ *Hanson Gregory, an American, claimed to have invented the ring-shaped doughnut in 1847 aboard a lime-trading ship when he was 16 years old*⋯ ⟨*Sep*⟩ *in 1847, vince invented rice-a-roni by adding a dry chicken soup mix to rice and macaroni*⋯ ⟨*Sep*⟩ *collectable tea cards were introduced in the 1950's as a way of advertising and retaining*⋯ ⟨*Sep*⟩ *it caused a sensation when it was introduced to the public in 1936*⋯ ⟨*Sep*⟩ *mind flashed to the summer session of 1891, when gulick introduced a new course in the psychology of play.* ⟨*/Doc*⟩ ⟨*Instruction*⟩ *Based on the above information and the given image, respond to this question with a short answer:* ⟨*Q*⟩ *In what year was this dessert first introduced?* ⟨*/Q*⟩ ⟨*/Instruction*⟩

Fig. B1 The image of the question: "in what year was this dessert first introduced?". (Colored figures are available in the online version at https://link.springer.com/journal/11633)

B.2 Training details

We initialize the retrieval module with ColBERTv2. We train the neural retriever with a learning rate of 1×10^{-4} and a batch size of 30. We fine-tune the LM beginning with the official checkpoint stage 2 of MiniGPT-v2. To train the LM, we employ an AdamW optimizer coupled with a cosine learning rate scheduler. Following the training process of MiniGPTv2, we freeze the ViT visual backbone in the architecture. We fine-tune the linear projection layer and the large language model (Llama-2-7B-Chat model) using LoRA. We set the rank $r=64$ and $\alpha=16$ for LoRA.

Appendix C. More qualitative examples

In Figs. C1–C4, we provide additional qualitative results.

Appendix D. Limitations and future work

In the E-step, we use the pseudo-label heuristic to approximate the posterior. This may cause the model to overemphasize passages that contain the answer keyword, while overlooking those that do not. We may consider more plausible heuristics and more effective ways to estimate the posterior.

Our approach may struggle in multi-hop questions. For example, "what part of this animal would be in use of it was playing the game that is played with items the man is holding?". In future work, we plan to further com-

Question: What kind of dog is this?
BootLM+MiniGPT-v2 : **schnauzer** ☒
Ground Truth: **terrier** ☑

Caption:
a small dog sitting on a rug with a donut in its mouth

Description:
The image features a black dog sitting on a white rug, with a pink donut toy in front of him. The dog is looking directly at the camera, and the donut toy is placed on the floor next to him. The dog is positioned in the center of the image, and the rug is visible in the background. The scene is well-lit, and the lighting is even throughout the room.

Passage 1	sassy yet charming, chihuahuas are tiny little dogs packed with tons of personality. they're considered to be one of the most popular mexican dog breeds\u2014and there are many reasons for it! with that perky little walk and graceful aura, chihuahuas are emotive and regal, seeking to always be at the center of your attention.
Passage 2	the miniature schnauzer dog breed has it all in one small package: intelligence, affection, an extroverted temperament, humor, and a personality that's twice as big as their bodies.
Passage 3	the chihuahua is the most famous of the \u201cpurse puppies,\u201d toy dogs toted around in chic upscale doggie bags by high-profile celebrities and socialites. the most famous celebrity chihuahua is tinker bell, who spends her days nestled in socialite paris hilton's handbag.
Passage 4	the chihuahua is one of the smallest dog breeds in the world. they are so small that they can fit inside of a purse or bag. some dogs measuring less than 6 inches are sometimes called teacup chihuahuas.
Passage 5	lively, courageous and alert chihuahuas are a small but mighty dog breed. learn more about chihuahua breed facts and information.
Passage 6	the miniature schnauzer is a breed of small dog of the schnauzer type that originated in germany in the mid-to-late 19th century. miniature schnauzers may have been developed from the smallest specimens of the standard schnauzer,[3][4] or crosses between the standard and one or more smaller breeds such as the affenpinscher, miniature pinscher, and poodles,[5][6][7][8] as farmers bred a small dog that was an efficient ratting dog.

Fig. C1 More qualitative results (Colored figures are available in the online version at https://link.springer.com/journal/11633)

Question: What are the headresses on these horses called?
BootLM+MiniGPT-v2 : **mask** ☒
Ground Truth: **bridle** ☑

Caption:
a man riding a horse with a mask on

Description:
The image features a man riding a horse wearing a helmet and a blue mask. The horse is also wearing a blue mask. The man is holding the reins of the horse, and the horse is standing on a dirt surface. There are other horses and people in the background, as well as a trailer in the distance.

Passage 1	a noseband is the part of a horse's **bridle** that encircles the nose and jaw of the horse. in english riding, where the noseband is separately attached to its own headstall or crownpiece, held independently of the bit, it is often called a cavesson or caveson noseband.
Passage 2	as defined in the oxford english dictionary, the \"**bridle**\" includes both the headstall that holds a bit that goes in the mouth of a horse, and the reins that are attached to the bit. headgear without a bit that uses a noseband to control a horse is called a hackamore, or, in some areas, a bitless bridle.
Passage 3	piece, there are a few parts that are universal to almost any bridle you are likely to run across. also called headstall and crownpiece is the strap that goes over the horse's head just behind their ears at the poll. the side of the horse's face along the cheekbone and attach to the bit.
Passage 4	a bearing rein, known today as an overcheck or a checkrein, is a piece of horse tack that runs from a point on the horse's back, over the head, to a bit. it is used to prevent the horse from lowering its head beyond a fixed point. the overcheck is generally not used to train dressage horses, because it can lead to the horse traveling with a hollow frame, the opposite of the rounded frame encouraged in dressage.
Passage 5	but in reality, horses wear blinders to help them perform their task better. 4 reasons horses wear blinders blinders, also called blinkers or winkers, are cups made from either leather or plastic. blinders, also called blinkers or winkers, are cups made from either leather or plastic. the blinders in racing are part of the horse's headgear.
Passage 6	a variation, called \"winkers\" are seen in australian thoroughbred horse racing. these are fleece rolls that are placed around the bridle cheek straps.

Fig. C2 More qualitative results (Colored figures are available in the online version at https://link.springer.com/journal/11633)

Question: Is this sport more popular in the US or the UK?
MiniGPT-v2: **us** ☒
BootLM+MiniGPT-v2 : **uk** ☑
Ground Truth: **uk** ☑

Caption:
a soccer player is kicking a ball in front of a crowd

Description:
The image depicts a soccer game in progress, with a player in the middle of the field attempting to kick the ball towards the goal. The player is wearing a black jersey and is surrounded by other players, including one who is standing near the goal. The ball is in mid-air, and the player is in mid-kick.There are several people watching the game from the stands, including a man in a red shirt and a woman in a white shirt. The stands are filled with spectators, and there are several people sitting on the bleachers. The scene is set in a soccer field, with a large crowd of people present. The stands are located on the sides of the field, and there are several people sitting on the bleachers.

Passage 1	football is an extremely popular sport in the united kingdom.
Passage 2	soccer is the most popular sport, anywhere but in the us.
Passage 3	many british sports have benefited from american influence and many american sports are becoming more popular in the uk. there's a lot of american influence within uk sport\u2026 in football, manchester united are owned by the glazer family and liverpool are owned by fenway sports group, who bought the club in 2010 from two other american owners \u2013 george gillett jr.
Passage 4	football (soccer) is the most popular sports in england, according to a top ten list from global sports 24/7. fishing does not make the top ten list. socce.
Passage 5	if you are based in the uk, you may well think football. undoubtedly, football will always be the most popular sport in that country, especially with the premier league being one of the most intense competitions in the world.
Passage 6	although football seems to run in every englishman's veins, there are also a lot of sports that flourish in england. rugby union is ranked as the second most popular sport in england.

Fig. C3 More qualitative results (Colored figures are available in the online version at https://link.springer.com/journal/11633)

Question: What occupation might he have?
MiniGPT-v2 : **construction** ☒
BootLM+MiniGPT-v2 : **forklift driver** ☑
Ground Truth: **forklift driver** ☑

Caption:
a toy truck with a man in the drivers seat

Description:
The image shows a small, vintage toy truck with a red body and a yellow and green trailer attached to it. The truck has a large, orange and white **forklift** on the back, which is being operated by a man in a brown and white outfit. The man is sitting on the forklift, with his hands on the steering wheel and his feet on the pedals. The truck is parked on a white surface, with a small table in the background.

Passage 1	delivery driver, driver, line haul driver, log truck driver, over the road driver (otr driver), production truck driver, road driver, semi truck driver, tractor trailer operator, truck driver view help what does this information tell me?this description is a quick overview of what workers in this career might do.\"also known as\" shows other common names for this career.what is the source of this information?this information comes from an o*net database. how much education do most people in this career have? find local training view help what does this information tell me?this chart shows you the range of education levels that people who currently work in this field have.
Passage 2	here's a quick look at the top ten most common jobs for former **forklift operators**: machine operator material handler warehouse associate warehouse worker driver order selector truck driver welder customer service representative maintenance technician these are all good jobs, all of them either utilizing some skill that a forklift operator would have or being a stepping stone to a different career. forklift operator would have or being a stepping stone to a different career. more recently, he's been quoted on usa today, businessinsider, and cnbc.
Passage 3	description: what do they do? operate one or several types of power construction equipment, such as motor graders, bulldozers, scrapers, compressors, pumps, derricks, shovels, tractors, or front-end loaders to excavate, move, and grade earth, erect structures, or pour concrete or other hard surface pavement. back hoe operator, engineering equipment operator, equipment operator (eo), forklift operator, heavy equipment operator, hot mix asphalt operator, machine operator, motor grader operator, operating engineer, track hoe operator view help what does this information tell me?this description is a quick overview of what workers in this career might do.\"also known as\" shows other common names for this career.what is the source of this information?this information comes from an o*net database.

Fig. C4 More qualitative results (Colored figures are available in the online version at https://link.springer.com/journal/11633)

pensate for this shortcoming through neural-symbolic methods.

Larger language models generally outperform smaller ones with greater capacity to learn and represent complex patterns in data[69]. They also benefit from exposure to more extensive training data, storing much more factual and commonsense knowledge. Future work may explore BootLM on more powerful and scalable model architectures.

References

[1] S. Antol, A. Agrawal, J. Lu, M. Mitchell, D. Batra, C. L. Zitnick, D. Parikh. VQA: Visual question answering. In *Proceedings of IEEE International Conference on Computer Vision*, Santiago, Chile, pp. 2425–2433, 2015. DOI:

10.1109/ICCV.2015.279.

[2] K. Marino, M. Rastegari, A. Farhadi, R. Mottaghi. OK-VQA: A visual question answering benchmark requiring external knowledge. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Long Beach, USA, pp. 3190–3199, 2019. DOI: 10.1109/CVPR.2019.00331.

[3] P. Wang, Q. Wu, C. Shen, A. Dick, A. Van Den Hengel. FVQA: Fact-based visual question answering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 10, pp. 2413–2427, 2017. DOI: 10.1109/TPAMI.2017.2754246.

[4] K. Marino, X. Chen, D. Parikh, A. Gupta, M. Rohrbach. KRISP: Integrating implicit and symbolic knowledge for open-domain knowledge-based VQA. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Nashville, USA, pp. 14106–14116, 2021. DOI: 10.1109/CVPR46437.2021.01389.

[5] J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Altenschmidt, S. Altman, S. Anadkat, R. Avila, I. Babuschkin, S. Balaji, V. Balcom, P. Baltescu, H. Bao, M. Bavarian, J. Belgum, I. Bello, J. Berdine, G. Bernadett-Shapiro, C. Berner, L. Bogdonoff, O. Boiko, M. Boyd, A. L. Brakman, G. Brockman, T. Brooks, M. Brundage, K. Button, T. Cai, R. Campbell, A. Cann, B. Carey, C. Carlson, R. Carmichael, B. Chan, C. Chang, F. Chantzis, D. Chen, S. Chen, R. Chen, J. Chen, M. Chen, B. Chess, C. Cho, C. Chu, H. W. Chung, D. Cummings, J. Currier, Y. Dai, C. Decareaux, T. Degry, N. Deutsch, D. Deville, A. Dhar, D. Dohan, S. Dowling, S. Dunning, A. Ecoffet, A. Eleti, T. Eloundou, D. Farhi, L. Fedus, N. Felix, S. P. Fishman, J. Forte, I. Fulford, L. Gao, E. Georges, C. Gibson, V. Goel, T. Gogineni, G. Goh, R. Gontijo-Lopes, J. Gordon, M. Grafstein, S. Gray, R. Greene, J. Gross, S. S. Gu, Y. Guo, C. Hallacy, J. Han, J. Harris, Y. He, M. Heaton, J. Heidecke, C. Hesse, A. Hickey, W. Hickey, P. Hoeschele, B. Houghton, K. Hsu, S. Hu, X. Hu, J. Huizinga, S. Jain, S. Jain, J. Jang, A. Jiang, R. Jiang, H. Jin, D. Jin, S. Jomoto, B. Jonn, H. Jun, T. Kaftan, Ł. Kaiser, A. Kamali, I. Kanitscheider, N. S. Keskar, T. Khan, L. Kilpatrick, J. W. Kim, C. Kim, Y. Kim, J. H. Kirchner, J. Kiros, M. Knight, D. Kokotajlo, Ł Kondraciuk, A. Kondrich, A. Konstantinidis, K. Kosic, G. Krueger, V. Kuo, M. Lampe, I. Lan, T. Lee, J. Leike, J. Leung, D. Levy, C. M. Li, R. Lim, M. Lin, S. Lin, M. Litwin, T. Lopez, R. Lowe, P. Lue, A. Makanju, K. Malfacini, S. Manning, T. Markov, Y. Markovski, B. Martin, K. Mayer, A. Mayne, B. McGrew, S. M. McKinney, C. McLeavey, P. McMillan, J. McNeil, D. Medina, A. Mehta, J. Menick, L. Metz, A. Mishchenko, P. Mishkin, V. Monaco, E. Morikawa, D. Mossing, T. Mu, M. Murati, O. Murk, D. Mély, A. Nair, R. Nakano, R. Nayak, A. Neelakantan, R. Ngo, H. Noh, L. Ouyang, C. O'Keefe, J. Pachocki, A. Paino, J. Palermo, A. Pantuliano, G. Parascandolo, J. Parish, E. Parparita, A. Passos, M. Pavlov, A. Peng, A. Perelman, F. de Avila Belbute Peres, M. Petrov, H. P. de Oliveira Pinto, M. Pokorny, M. Pokrass, V. H. Pong, T. Powell, A. Power, B. Power, E. Proehl, R. Puri, A. Radford, J. Rae, A. Ramesh, C. Raymond, F. Real, K. Rimbach, C. Ross, B. Rotsted, H. Roussez, N. Ryder, M. Saltarelli, T. Sanders, S. Santurkar, G. Sastry, H. Schmidt, D. Schnurr, J. Schulman, D. Selsam, K. Sheppard, T. Sherbakov, J. Shieh, S. Shoker, P. Shyam, S. Sidor, E. Sigler, M. Simens, J. Sitkin, K. Slama, I. Sohl, B. Sokolowsky, Y. Song, N. Staudacher, F. P. Such, N. Summers, I. Sutskever, J. Tang, N. Tezak, M. B. Thompson, P. Tillet, A. Tootoonchian, E. Tseng, P. Tuggle, N. Turley, J. Tworek, J. F. C. Uribe, A. Vallone, A. Vijayvergiya, C. Voss, C. Wainwright, J. J. Wang, A. Wang, B. Wang, J. Ward, J. Wei, C. J. Weinmann, A. Welihinda, P. Welinder, J. Weng, L. Weng, M. Wiethoff, D. Willner, C. Winter, S. Wolrich, H. Wong, L. Workman, S. Wu, J. Wu, M. Wu, K. Xiao, T. Xu, S. Yoo, K. Yu, Q. Yuan, W. Zaremba, R. Zellers, C. Zhang, M. Zhang, S. Zhao, T. Zheng, J. Zhuang, W. Zhuk, B. Zoph, GPT-4 technical report, [Onine], Available: https://arxiv.org/abs/2303.08774, 2024.

[6] J. Chen, D. Zhu, X. Shen, X. Li, Z. Liu, P. Zhang, R. Krishnamoorthi, V. Chandra, Y. Xiong, M. Elhoseiny. MiniGPT-v2: Large language model as a unified interface for vision-language multi-task learning, [Online], Available: https://arxiv.org/abs/2310.09478, 2023.

[7] H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, N. Bashlykov, S. Batra, P. Bhargava, S. Bhosale, D. Bikel, L. Blecher, C. C. Ferrer, M. Chen, G. Cucurull, D. Esiobu, J. Fernandes, J. Fu, W. Fu, B. Fuller, C. Gao, V. Goswami, N. Goyal, A. Hartshorn, S. Hosseini, R. Hou, H. Inan, M. Kardas, V. Kerkez, M. Khabsa, I. Kloumann, A. Korenev, P. S. Koura, M. A. Lachaux, T. Lavril, J. Lee, D. Liskovich, Y. Lu, Y. Mao, X. Martinet, T. Mihaylov, P. Mishra, I. Molybog, Y. Nie, A. Poulton, J. Reizenstein, R. Rungta, K. Saladi, A. Schelten, R. Silva, E. M. Smith, R. Subramanian, X. E. Tan, B. Tang, R. Taylor, A. Williams, J. X. Kuan, P. Xu, Z. Yan, I. Zarov, Y. Zhang, A. Fan, M. Kambadur, S. Narang, A. Rodriguez, R. Stojnic, S. Edunov, T. Scialom. Llama 2: Open foundation and fine-tuned chat models, [Online], Available: https://arxiv.org/abs/2307.09288, 2023.

[8] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, D. Amodei. Language models are few-shot learners. In *Proceedings of the 34th Conference on Neural Information Processing Systems*, pp. 1877–1901, 2020.

[9] Z. Shao, Z. Yu, M. Wang, J. Yu. Prompting large language models with answer heuristics for knowledge-based visual question answering. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Vancouver, Canada, pp. 14974–14983, 2023. DOI: 10.1109/CVPR52729.2023.01438.

[10] Y. Hu, H. Hua, Z. Yang, W. Shi, N. A. Smith, J. Luo. PromptCap: Prompt-guided task-aware image captioning, [Online], Available: https://arxiv.org/abs/2211.09699, 2023.

[11] Z. Yang, Z. Gan, J. Wang, X. Hu, Y. Lu, Z. Liu, L. Wang. An empirical study of GPT-3 for few-shot knowledge-based VQA. In *Proceedings of the 36th AAAI Conference on Artificial Intelligence*, pp. 3081–3089, 2022. DOI: 10.1609/aaai.v36i3.20215.

[12] Y. Lin, Y. Xie, D. Chen, Y. Xu, C. Zhu, L. Yuan. REVIVE: Regional visual representation matters in knowledge-based visual question answering. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, New Orleans, USA, Article number 767, 2022.

[13] L. Huang, W. Yu, W. Ma, W. Zhong, Z. Feng, H. Wang, Q. Chen, W. Peng, X. Feng, B. Qin, T. Liu. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Transactions on Information Systems*, vol. 43, no. 2, Article number 42, 2025. DOI: 10.1145/3703155.

[14] W. Lin, J. Chen, J. Mei, A. Coca, B. Byrne. Fine-grained late-interaction multi-modal retrieval for retrieval augmented visual question answering. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, New Orleans, USA, Article number 990, 2024. DOI: 10.5555/3666122.3667112.

[15] N. D. Huynh, M. R. Bouadjenek, S. Aryal, I. Razzak, H. Hacid. Visual question answering: From early developments to recent advances – a survey, [Online], Available: https://arxiv.org/abs/2501.03939, 2025.

[16] L. Ma, Z. Lu, H. Li. Learning to answer questions from image using convolutional neural network. In *Proceedings of the 30th AAAI Conference on Artificial Intelligence*, Phoenix, USA, pp. 3567–3573, 2016. DOI: 10.1609/aaai.v30i1.10442.

[17] M. Malinowski, M. Rohrbach, M. Fritz. Ask your neurons: A neural-based approach to answering questions about images. In *Proceedings of IEEE International Conference on Computer Vision*, Santiago, Chile, 2015. DOI: 10.1109/ICCV.2015.9.

[18] Z. Yang, X. He, J. Gao, L. Deng, A. Smola. Stacked attention networks for image question answering. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, Las Vegas, USA, pp. 21–29, 2016. DOI: 10.1109/CVPR.2016.10.

[19] H. Xu, K. Saenko. Ask, attend and answer: Exploring question-guided spatial attention for visual question answering. In *Proceedings of the 14th European Conference on Computer Vision*, Amsterdam, The Netherlands, pp. 451–466, 2016. DOI: 10.1007/978-3-319-46478-7_28.

[20] J. Lu, J. Yang, D. Batra, D. Parikh. Hierarchical question-image co-attention for visual question answering. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, Barcelona, Spain, pp. 289–297, 2016. DOI: 10.5555/3157096.3157129.

[21] Z. Yu, J. Yu, Y. Cui, D. Tao, Q. Tian. Deep modular co-attention networks for visual question answering. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Long Beach, USA, pp. 6274–6283, 2019. DOI: 10.1109/CVPR.2019.00644.

[22] J. Andreas, M. Rohrbach, T. Darrell, D. Klein. Neural module networks. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, Las Vegas, USA, pp. 39–48, 2016. DOI: 10.1109/CVPR.2016.12.

[23] R. Hu, J. Andreas, M. Rohrbach, T. Darrell, K. Saenko. Learning to reason: End-to-end module networks for visual question answering. In *Proceedings of IEEE International Conference on Computer Vision*, Venice, Italy, pp. 804–813, 2017. DOI: 10.1109/ICCV.2017.93.

[24] R. Hu, J. Andreas, T. Darrell, K. Saenko. Explainable neural computation via stack neural module networks. In *Proceedings of the 15th European Conference on Computer Vision*, Munich, Germany, pp. 55–71, 2018. DOI: 10.1007/978-3-030-01234-2_4.

[25] D. Teney, L. Liu, A. van den Hengel. Graph-structured representations for visual question answering. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, Honolulu, USA, pp. 3233–3241, 2017. DOI: 10.1109/CVPR.2017.344.

[26] M. Narasimhan, S. Lazebnik, A. G. Schwing. Out of the box: Reasoning with graph convolution nets for factual visual question answering. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, Montreal, Canada, pp. 2659–2670, 2018. DOI: 10.5555/3327144.3327190.

[27] D. Guo, C. Xu, D. Tao. Bilinear graph networks for visual question answering. *IEEE Transactions on Neural Networks and Learning Systems*, vol. 34, no. 2, pp. 1023–1034, 2023. DOI: 10.1109/TNNLS.2021.3104937.

[28] X. Li, X. Yin, C. Li, P. Zhang, X. Hu, L. Zhang, L. Wang, H. Hu, L. Dong, F. Wei, Y. Choi, J. Gao. OSCAR: Object-semantics aligned pre-training for vision-language tasks. In *Proceedings of the 16th European Conference on Computer Vision*, Glasgow, UK, pp. 121–137, 2020. DOI: 10.1007/978-3-030-58577-8_8.

[29] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, I. Sutskever. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning*, pp. 8748–8763, 2021.

[30] L. Yuan, D. Chen, Y. L. Chen, N. Codella, X. Dai, J. Gao, H. Hu, X. Huang, B. Li, C. Li, C. Liu, M. Liu, Z. Liu, Y. Lu, Y. Shi, L. Wang, J. Wang, B. Xiao, Z. Xiao, J. Yang, M. Zeng, L. Zhou, P. Zhang. Florence: A new foundation model for computer vision, [Online], Available: https://arxiv.org/abs/2111.11432, 2021.

[31] Z. Wang, J. Yu, A. W. Yu, Z. Dai, Y. Tsvetkov, Y. Cao. SimVLM: Simple visual language model pretraining with weak supervision. In *Proceedings of the 10th International Conference on Learning Representations*, 2022.

[32] P. Zhang, X. Li, X. Hu, J. Yang, L. Zhang, L. Wang, Y. Choi, J. Gao. VinVL: Revisiting visual representations in vision-language models. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Nashville, USA, pp. 5575–5584, 2021. DOI: 10.1109/CVPR46437.2021.00553.

[33] W. Li, C. Gao, G. Niu, X. Xiao, H. Liu, J. Liu, H. Wu, H. Wang. UNIMO: Towards unified-modal understanding and generation via cross-modal contrastive learning. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*, pp. 2592–2607, 2021.

[34] P. Wang, Q. Wu, C. Shen, A. Dick, A. van den Hengel. Explicit knowledge-based reasoning for visual question answering. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, Melbourne, Australia, pp. 1290–1296, 2017. DOI: 10.24963/ijcai.2017/179.

[35] M. Narasimhan, A. G. Schwing. Straight to the facts: Learning knowledge base retrieval for factual visual question answering. In *Proceedings of the 15th European Conference on Computer Vision*, Munich, Germany, pp. 451–468, 2018. DOI: 10.1007/978-3-030-01237-3_28.

[36] S. Shah, A. Mishra, N. Yadati, P. P. Talukdar. KVQA: Knowledge-aware visual question answering. In *Proceedings of the 33rd AAAI Conference on Artificial Intelligence*, Honolulu, USA, pp. 8876–8884, 2019. DOI: 10.1609/aaai.v33i01.33018876.

[37] J. Wu, J. Lu, A. Sabharwal, R. Mottaghi. Multi-modal answer validation for knowledge-based VQA. In *Proceedings of the 36th AAAI Conference on Artificial Intelligence*, pp. 2712–2721, 2022. DOI: 10.1609/aaai.v36i3.20174.

[38] X. Xing, M. Liang, Y. Wu. TOA: Task-oriented active VQA. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, New Orleans, USA, Article number 2353, 2023. DOI: 10.5555/3666122.3668475.

[39] L. Gui, B. Wang, Q. Huang, A. Hauptmann, Y. Bisk, J. Gao. KAT: A knowledge augmented transformer for vision-and-language. In *Proceedings of Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Seattle, USA, pp. 956–968, 2022. DOI: 10.18653/v1/2022.naacl-main.70.

[40] M. Luo, Y. Zeng, P. Banerjee, C. Baral. Weakly-supervised visual-retriever-reader for knowledge-based question answering. In *Proceedings of Conference on Empirical Methods in Natural Language Processing*, Punta Cana,

Dominican Republic, pp. 6417–6431, 2021. DOI: 10.18653/v1/2021.emnlp-main.517.

[41] F. Gao, Q. Ping, G. Thattai, A. Reganti, Y. N. Wu, P. Natarajan. Transform-retrieve-generate: Natural language-centric outside-knowledge visual question answering. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, New Orleans, USA, pp. 5057–5067, 2022. DOI: 10.1109/CVPR52688.2022.00501.

[42] Y. Ding, J. Yu, B. Liu, Y. Hu, M. Cui, Q. Wu. MuKEA: Multimodal knowledge extraction and accumulation for knowledge-based visual question answering. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, New Orleans, USA, pp. 5079–5089, 2022. DOI: 10.1109/CVPR52688.2022.00503.

[43] W. Dai, J. Li, D. Li, A. M. H. Tiong, J. Zhao, W. Wang, B. Li, P. Fung, S. Hoi. InstructBLIP: Towards general-purpose vision-language models with instruction tuning. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, New Orleans, USA, Article number 2142, 2023.

[44] H. Liu, C. Li, Q. Wu, Y. J. Lee. Visual instruction tuning. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, New Orleans, USA, Article number 1516, 2023. DOI: 10.5555/3666122.3667638.

[45] D. Driess, F. Xia, M. S. M. Sajjadi, C. Lynch, A. Chowdhery, B. Ichter, A. Wahid, J. Tompson, Q. Vuong, T. Yu, W. Huang, Y. Chebotar, P. Sermanet, D. Duckworth, S. Levine, V. Vanhoucke, K. Hausman, M. Toussaint, K. Greff, A. Zeng, I. Mordatch, P. Florence. PaLM-E: An embodied multimodal language model. In *Proceedings of the 40th International Conference on Machine Learning*, Honolulu, USA, pp. 8469–8488, 2023.

[46] J. B. Alayrac, J. Donahue, P. Luc, A. Miech, I. Barr, Y. Hasson, K. Lenc, A. Mensch, K. Millican, M. Reynolds, A. Miech, I. Barr, Y. Hasson, K. Lenc, A. Mensch, K. Millicah, M. Reynolds, R. Ring, E. Rutherford, S. Cabi, T. Han, Z. Gong, S. Samangooei, M. Monteiro, J. Menick, S. Borgeaud, A. Brock, A. Nematzadeh, S. Sharifzadeh, M. Binkowski, R. Barreira, O. Vinyals, A. Zisserman, K. Simonyan. Flamingo: A visual language model for few-shot learning. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, New Orleans, USA, Article number 1723, 2022. DOI: 10.5555/3600270.3601993.

[47] M. Yasunaga, A. Aghajanyan, W. Shi, R. James, J. Leskovec, P. Liang, M. Lewis, L. Zettlemoyer, W. T. Yih. Retrieval-augmented multimodal language modeling. In *Proceedings of the 40th International Conference on Machine Learning*, Honolulu, USA, pp. 39755–39769, 2022. DOI: 10.48550/arXiv.2211.12561.

[48] H. Wang, W. Ge. Q&A prompts: Discovering rich visual clues through mining question-answer prompts for VQA requiring diverse world knowledge. In *Proceedings of the 18th European Conference on Computer Vision*, Milan, Italy, pp. 274–292, 2024. DOI: 10.1007/978-3-031-72946-1_16.

[49] Z. Ma, S. Li, B. Sun, J. Cai, Z. Long, F. Ma. GeReA: Question-aware prompt captions for knowledge-based visual question answering, [Online], Available: https://arxiv.org/abs/2402.02503, 2024.

[50] V. Karpukhin, B. Oğuz, S. Min, P. Lewis, L. Wu, S. Edunov, D. Chen, W. T. Yih. Dense passage retrieval for open-domain question answering. In *Proceedings of Conference on Empirical Methods in Natural Language Processing*, pp. 6769–6781, 2020. DOI: 10.18653/v1/2020.emnlp-main.550.

[51] J. Wu, R. Mooney. Entity-focused dense passage retrieval for outside-knowledge visual question answering. In *Proceedings of Conference on Empirical Methods in Natural Language Processing*, Abu Dhabi, UAE, pp. 8061–8072, 2022. DOI: 10.18653/v1/2022.emnlp-main.551.

[52] W. Lin, B. Byrne. Retrieval augmented visual question answering with outside knowledge. In *Proceedings of Conference on Empirical Methods in Natural Language Processing*, Abu Dhabi, UAE, pp. 11238–11254, 2022. DOI: 10.18653/v1/2022.emnlp-main.772.

[53] K. Santhanam, O. Khattab, J. Saad-Falcon, C. Potts, M. Zaharia. ColBERTv2: Effective and efficient retrieval via lightweight late interaction. In *Proceedings of Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Seattle, USA, pp. 3715–3734, 2022. DOI: 10.18653/v1/2022.naacl-main.272.

[54] J. Rao, Z. Shan, L. Liu, Y. Zhou, Y. Yang. Retrieval-based knowledge augmented vision language pre-training. In *Proceedings of the 31st ACM International Conference on Multimedia*, Ottawa, Canada, pp. 5399–5409, 2023. DOI: 10.1145/3581783.3613848.

[55] Y. Gao, Y. Xiong, X. Gao, K. Jia, J. Pan, Y. Bi, Y. Dai, J. Sun, H. Wang, H. Wang. Retrieval-augmented generation for large language models: A survey, [Online], Available: https://arxiv.org/abs/2312.10997, 2024.

[56] W. Chen, H. Hu, X. Chen, P. Verga, W. W. Cohen. MuRAG: Multimodal retrieval-augmented generator for open question answering over images and text. In *Proceedings of Conference on Empirical Methods in Natural Language Processing*, Abu Dhabi, UAE, pp. 5558–5570, 2022. DOI: 10.18653/v1/2022.emnlp-main.375.

[57] C. Li, Z. Li, C. Jing, S. Liu, W. Shao, Y. Wu, P. Luo, Y. Qiao, K. Zhang. SearchLVLMs: A plug-and-play framework for augmenting large vision-language models by searching up-to-date internet knowledge. In *Proceedings of the 38th International Conference on Neural Information Processing Systems*, Vancouver, Canada, Article number 2060, 2025. DOI: 10.5555/3737916.3739976.

[58] C. Tan, J. Wei, L. Sun, Z. Gao, S. Li, B. Yu, R. Guo, S. Z. Li. Retrieval meets reasoning: Even high-school textbook knowledge benefits multimodal reasoning, [Online], Available: https://arxiv.org/abs/2405.20834, 2024.

[59] Z. Chen, C. Xu, Y. Qi, J. Guo. MLLM is a strong reranker: Advancing multimodal retrieval-augmented generation via knowledge-enhanced reranking and noise-injected training, [Online], Available: https://arxiv.org/abs/2407.21439, 2024.

[60] O. Khattab, M. Zaharia. ColBERT: Efficient and effective passage search via contextualized late interaction over BERT. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 39–48, 2020. DOI: 10.1145/3397271.3401075.

[61] R. Speer, J. Chin, C. Havasi. ConceptNet 5.5: An open multilingual graph of general knowledge. In *Proceedings of the 31st AAAI Conference on Artificial Intelligence*, San Francisco, USA, pp. 4444–4451, 2017. DOI: 10.1609/aaai.v31i1.11164.

[62] N. Tandon, G. de Melo, G. Weikum. WebChild 2.0: Fine-grained commonsense knowledge distillation. In *Proceedings of ACL, System Demonstrations*, Association for Computational Linguistics, Vancouver, Canada, pp. 115–120, 2017. DOI: 10.18653/v1/P17-4020.

[63] S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, Z. Ives. DBpedia: A nucleus for a web of open data. In *Proceedings of the 6th International Semantic Web Confer-*

ence, the 2nd Asian Semantic Web Conference, Busan, Republic of Korea, pp. 722–735, 2007. DOI: 10.1007/978-3-540-76298-0_52.

[64] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, P. J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, vol. 21, no. 1, Article number 140, 2020. DOI: 10.5555/3455716.3455856.

[65] J. Li, D. Li, S. Savarese, S. Hoi. BLIP-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *Proceedings of the 40th International Conference on Machine Learning*, Honolulu, USA, pp. 19730–19742, 2023.

[66] X. Chen, X. Wang, S. Changpinyo, A. J. Piergiovanni, P. Padlewski, D. Salz, S. Goodman, A. Grycner, B. Mustafa, L. Beyer, A. Kolesnikov, J. Puigcerver, N. Ding, K. Rong, H. Akbari, G. Mishra, L. Xue, A. V. Thapliyal, J. Bradbury, W. Kuo. PaLI: A jointly-scaled multilingual language-image model. In *Proceedings of the 11th International Conference on Learning Representations*, Kigali, Rwanda, 2023.

[67] F. Gardères, M. Ziaeefard, B. Abeloos, F. Lecue. ConceptBert: Concept-aware representation for visual question answering. In *Proceedings of Findings of the Association for Computational Linguistics*, pp. 489–498, 2020. DOI: 10.18653/v1/2020.findings-emnlp.44.

[68] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, W. Chen. LoRA: Low-rank adaptation of large language models. In *Proceedings of the 10th International Conference on Learning Representations*, 2022.

[69] J. Kaplan, S. McCandlish, T. Henighan, T. B. Brown, B. Chess, R. Child, S. Gray, A. Radford, J. Wu, D. Amodei. Scaling laws for neural language models, [Online], Available: https://arxiv.org/abs/2001.08361, 2020.

Yanze Min received the B. Eng. degree in computer science and technology from Tsinghua University, China in 2016. He is currently a Ph. D. degree candidate in Department of Computer Science and Technology, Tsinghua University, China.

His research interest is machine learning reasoning methods with domain knowledge, including structural deep generative models, symbolic and logical deduction in deep neural networks, and Bayesian inference methods.

E-mail: minyz16@mails.tsinghua.edu.cn (Corresponding author)

ORCID iD: 0009-0002-2470-8399

Yawei Sun received the Ph.D. degree in computer science from Nanjing University, China in 2022. He is currently a researcher collaborating with the Tsinghua machine learning research group at Tsinghua University, China.

His research interests include knowledge retrieval-augmented large models, knowledge-integrated model robustness, and knowledge graph construction.

E-mail: ywsun.nju@gmail.com

Yin Zhu received the B. Sc. and M. Eng. degrees in computer science and technology from Nanjing University, China in 2020 and 2024, respectively. He is currently working at Alibaba Inc, China.

His research interests include text generation and language model agent.

E-mail: yinzhu@smail.nju.edu.cn

Jun Zhu received the B. Eng., M. Sc. and Ph. D. degrees in computer science from Tsinghua University, China in 2005, 2007 and 2009, respectively. He is a Bosch AI professor in the Department of Computer Science and Technology at Tsinghua University, China. He is an IEEE/AAAI Fellow. He was an adjunct faculty at Carnegie Mellon University, USA from 2015 to 2018. He has published over 100 peer-reviewed papers in the prestigious conferences and journals, including ICML, NeurIPS, KDD, *Journal of Machine Learning Research*, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, etc. He is an Associate Editor-in-Chief for *IEEE Transactions on Pattern Analysis and Machine Intelligence*, and Associate Editors for *Artificial Intelligence*, *Acta Automatica Sinica* and *Machine Intelligence Research*. He served as area chair/senior PC for ICML (2014–2021), NeurIPS (2013, 2015, 2018, 2019, 2020), IJCAI (2013–2021), UAI (2014–2019), and AAAI (2016, 2017, 2019, 2020, 2021). He was a local co-chair of ICML2014 and workshop co-chair of ICML2021. He is a recipient of Microsoft Fellowship (2007), CCF Distinguished PhD Thesis Award (2009), IEEE Intelligent Systems "AI's 10 to Watch" Award (2013), NSFC Excellent Young Scholar Award (2013), CCF Young Scientist Award (2013), MIT TR35 China (2017), ICME 2018 best paper award, and XPlorer Prize (2020).

His research interests include the development of statistical machine learning methods for solving scientific and engineering problems arising from artificial learning, reasoning, and decision-making in the dynamic worlds.

E-mail: dcszj@mail.tsinghua.edu.cn

Bo Zhang graduated from Department of Automatic Control, Tsinghua University, China. He is now a professor of Department of Computer Science and Technology, Tsinghua University, China, a Fellow of Chinese Academy of Sciences, and the Director of Institute for Artificial Intelligence, Tsinghua University, China. He received the Honorary Doctor of Natural Sciences by Hamburg University, Germany in 2011, and the lifetime achievement award by China Computer Federation (CCF) in 2014, Wu Wenjun Artificial Intelligence the Highest Achievement Award (2019).

His research interests include artificial intelligence, artificial neural networks, genetic algorithms, intelligent robotics, pattern recognition and intelligent control.

E-mail: dcszb@mail.tsinghua.edu.cn

Machine Intelligence Research

Citation: Y. Min, Y. Sun, Y. Zhu, J. Zhu, B. Zhang. Bootstrapping large language models with outside–knowledge for knowledge–based visual question answering. *Machine Intelligence Research*, vol.23, no.1, pp.115–132, 2026. https://doi.org/10.1007/s11633–025–1591–z