Weakly-Supervised Visual-Retriever-Reader Pipeline for Knowledge-Based VQA Tasks

by

Yankai Zeng

A Thesis Presented in Partial Fulfillment
of the Requirements for the Degree
Master of Science

Approved June 2021 by the
Graduate Supervisory Committee:

Chitta Baral, Chair
Yezhou Yang
Samira Ghayekhloo

ARIZONA STATE UNIVERSITY

August 2021

ABSTRACT

Visual question answering (VQA) is a task that answers the questions by giving an image, and thus involves both language and vision methods to solve, which make the VQA tasks a frontier interdisciplinary field. In recent years, as the great progress made in simple question tasks (e.g. object recognition), researchers start to shift their interests to the questions that require knowledge and reasoning.

Knowledge-based VQA requires answering questions with external knowledge in addition to the content of images. One dataset that is mostly used in evaluating knowledge-based VQA is OK-VQA, but it lacks a gold standard knowledge corpus for retrieval. Existing work leverages different knowledge bases (e.g., ConceptNet and Wikipedia) to obtain external knowledge. Because of varying knowledge bases, it is hard to fairly compare models' performance. To address this issue, this paper collects a natural language knowledge base that can be used for any question answering (QA) system.

Moreover, a Visual Retriever-Reader pipeline is proposed to approach knowledge-based VQA, where the visual retriever aims to retrieve relevant knowledge, and the visual reader seeks to predict answers based on given knowledge. The retriever is constructed with two versions: term based retriever which uses best matching 25 (BM25), and neural based retriever where the latest dense passage retriever (DPR) is introduced. To encode the visual information, the image and caption are encoded separately in the two kinds of neural based retriever: Image-DPR and Caption-DPR. There are also two styles of readers, classification reader and extraction reader. Both the retriever and reader are trained with weak supervision. The experimental results show that a good retriever can significantly improve the reader's performance on the OK-VQA challenge.

DEDICATION

I dedicate this work to my family, who have been supporting me to pull through.

ACKNOWLEDGMENTS

Through my thesis, I have received a great amount of support from my instructor, mentor, and other enthusiastic lab colleagues.

I wish to express my deepest gratitude to my instructor, Dr. Chitta Baral, for instructing me on the Natural Language Processing fields and enlighten my interest in the Visual Question Answering challenges. He precisely pointed out the deficiencies during my work and provided some advice that direct me.

I also would like to thank my mentor, Man Luo. She provided me the idea to construct the model and did a significant effort in the model training part. Without her, the performance of our model can never beat the state-of-the-art in the OK-VQA task so quickly. I also learned many tricks and skills from her, especially in deep learning programming.

I would also thank the assistantship from the lab colleagues. I wish to express my grateful thank to Shailaja Sampat for inspiring me to conquer the OK-VQA challenge; Pratyay Banerjee for his ideas greatly help improve the performance; Saadat Anwar, Ming Shen, Soujanya Ranganatha for their help in collecting knowledge.

TABLE OF CONTENTS

Page

CHAPTER Page

CHAPTER Page

LIST OF TABLES

LIST OF FIGURES

Chapter 1

INTRODUCTION

Over the years, people's interests keep growing in making computers to answer questions like humans, and question answering (QA) has indubitably become a high-profile domain in natural language processing (NLP). Among the various QAs, Visual Question Answering (VQA) ranks as one of the most challenging tasks as it requires combining the visual and linguistic information to answer the question. This work targets to address the knowledge-based VQA, where knowledge present in an image is insufficient to answer a question, and thus the external knowledge is required.

1.1 Motivation

1.1.1 OK-VQA Challenge

OK-VQA (Marino et al. 2019) is a knowledge-based VQA dataset proposed recently. Figure 1 shows two examples from the OK-VQA benchmark. In each of the two examples, external knowledge is needed to answer the question. For instance, in the first example, to identify the vehicle used in the item shown in the image (top-left), a system needs first ground the referred object to a fire hydrant and then seek external knowledge (an example is top-right of the image). Compared to other knowledge-based VQA tasks that generate questions based on some knowledge source and (or) use question templates (such as FVQA (P. Wang et al. 2017) and KB-VQA (P. Wang et al. 2015)), OK-VQA generated natural questions (details see Table 1).

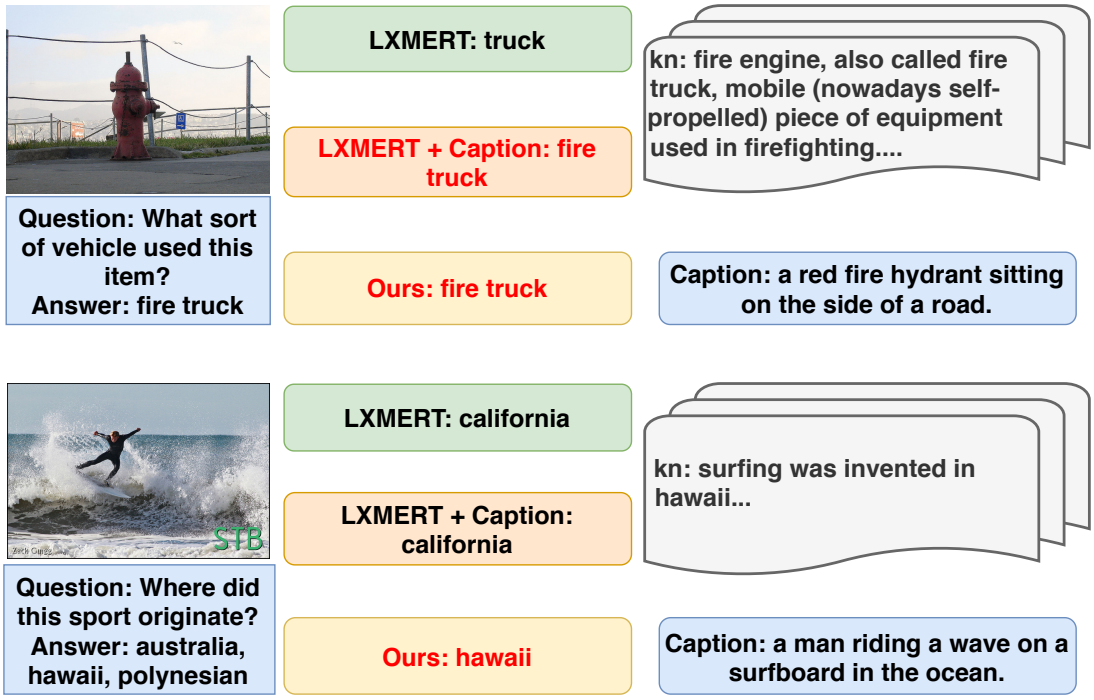

Figure 1. Two examples from OK-VQA.

Note: The middle column are predictions by two baselines and one by our proposed Visual-Retriever-Reader pipeline. The left column are relevant knowledge and the corresponding captioning of images.

More essentially, OK-VQA requires knowledge in rich diversity, where the topics are divided into 11 classes, and thus this task is more challenging. The 11 classes are listed as below:

1. Brands, Companies and Products
2. Plants and Animals
3. Science and Technology
4. Sports and Recreation
5. Vehicles and Transportation
6. Objects, Material and Clothing

7. Geography, History, Language and Culture
8. Weather and Climate
9. People and Everyday life
10. Cooking and Food
11. Other

Based on the features mentioned above, here we use the OK-VQA Dataset as the basis to start our further work on knowledge-based VQA.

Table 1. Comparison of different knowledge-based VQAs

Dataset	**# I**	**# Q**	**Image**	**Knowledge**	**KB Type**	**Template**
KB-VQA	700	2,402	COCO & IN	Human	-	yes
FVQA	2,190	5,826	COCO	DB & CN & WC	KG	yes
KVQA	24,602	183,307	Wiki	Wiki	KG	paraphrased
OK-VQA	14,031	14,055	COCO	Human	-	no
text-KVQA	257,380	1,322,272	GSI	Wiki & IMDb & BC	KG	paraphrased

Note: Here # I denotes the number of images whereas # Q denotes the number of questions. Image refers to the image source, Knowledge refers to the knowledge source to generate questions, KB Type shows the type of the knowledge base provided to answer these questions, and Template indicates whether the template is used to generate the questions. Abbreviations: IN-ImageNet, Wiki-Wikidata, DB-DBpedia, CN-ConceptNet, WC-WebChild, KG-knowledge graph, GSI-Google image search, BC-a book catalogue (Iwana et al. 2016)

1.1.2 Create A Knowledge Base for OK-VQA

Although the OK-VQA benchmark encourages a VQA system to rely on external resources to answer the question, it does not provide a knowledge corpus for a QA system to use. Some existing methods utilize different resources such as ConceptNet (Speer, Chin, and Havasi 2017), WordNet (Miller 1995), and Wikidata (Vrandečić and Krötzsch 2014), but consequently bring about the following issues:

Q: When was the cola brand on the sign founded?
A: 1892

Wikidata

In 1892, Candler set out to incorporate a second company; "The Coca-Cola Company" (the current corporation) ...

ConceptNet

<Cola, RelatedTo, Limonade>, <diet coke, RelatedTo, cola> <Coca Cola, IsA, Coke>, <water, RelatedTo, Cola> ...

Figure 2. Example for Different Knowledge Structures

Note: The differences between the natural language knowledge base Wikidata and knowledge-graph structured ConceptNet. Here the ConceptNet knowledge is constructed by relation triple "< Object, Relation, Subject >".

1. It is difficult to fairly compare different VQA systems as it is unclear whether the difference in performance arises from differing model architectures or the different knowledge sources.
2. Most current knowledge bases have different knowledge format, such as the structured ConceptNet and the unstructured Wikipedia (See Figure 2), demand different modules to retrieve knowledge, resulting in making a knowledge-based VQA system complicated.

Therefore, there is a need for a general and easy-to-use knowledge base for OK-VQA task.

1.1.3 Natural Language Knowledge VS. Knowledge Graph

Most of the current knowledge-based VQA tasks (see Table 1) provide a knowledge graph (KG) based structured knowledge set. Truly, the structured knowledge is

friendly for computer to reason, but there are still some limitations compared to the unstructured knowledge.

1. Natural language knowledge (NLK) is easy to obtain, as it is widely used and constitutes our daily conversations, production, commerce and all other activities, but structured knowledge sources can only cover a limited amount of knowledge. For example, ConceptNet provides only 34 relation types.
2. There is a vast amount of knowledge that is hard to be described by a relation in a knowledge graph, such as, *describe the logo of Apple Inc.* However, with natural language, it is simple to be depicted as "An apple with a bite taken."
3. Constructing a structured KG requires using NLP techniques like parsing, where the reliability is depend on the parser. Therefore, each structured knowledge base is generated with large amount of human annotation. On the contrary, unlike KG, no more further processing required for NLK.
4. The existing retrieval methods that widely used in information retrieval (IR) field can be easily applied to NLK to retrieve a relevant knowledge. As the neural network models has achieved a great progress in IR area (Lee, Chang, and Toutanova 2019; Karpukhin et al. 2020), it can also improve the performance of the knowledge-based VQA tasks.

In consideration of above aspects, we decide to build the knowledge base in unstructured natural language style.

1.2 Contribution

The contributions are three folds. First, we build a general easy-to-use knowledge corpus for the OK-VQA benchmark, which makes model evaluation fair. Second, we

propose a Visual-Retriever-Reader pipeline adapted from the NLP domain for the knowledge-based VQA task. Our model establishes a new state-of-the-art. Third, our experiments reveal several insights as mentioned above, and open a new research direction.

1.2.1 Knowledge Collection

We collect a knowledge corpus for the OK-VQA benchmark. The corpus is automatically collected via Google Search[1] by using the training-split question and the corresponding answers. The details of the collection will be shown in Chapter 3 We also provide a training corpus with 112,724 knowledge sentences in total. The knowledge corpus is in a uniform format, i.e., natural language. Thus it is easy to use by other OK-VQA methods. As we will show in Chapter 6, the knowledge base provides rich information to answer OK-VQA questions.

1.2.2 Retriever-Reader Pipeline

Utilizing the curated corpus, we further develop a weakly-supervised Visual-Retriever-Reader and evaluate it on the OK-VQA challenge. It consists of two stages as seen in Figure 3. In the first stage, the visual retriever retrieves relevant knowledge from the corpus. In the second stage, the visual reader predicts an answer based on the given knowledge. Such a pipeline is well-studied in text-only open-domain QA (Chen et al. 2017; Karpukhin et al. 2020). We apply its principles to the multi-modal vision and language domain with novel adaptations. On the retriever side, we introduce

[1]https://developers.google.com/custom-search/v1/

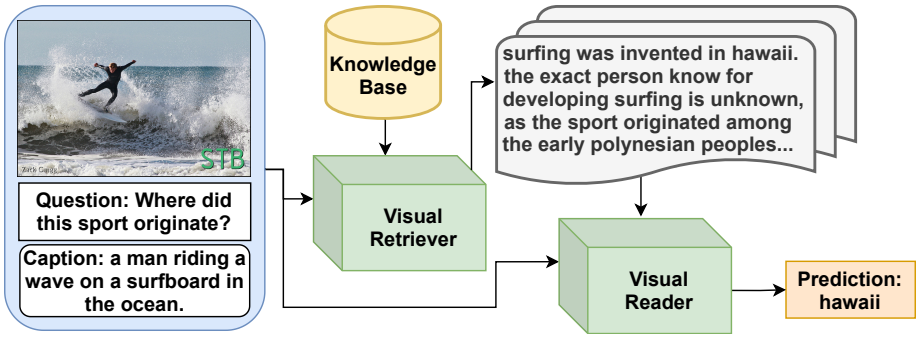

Figure 3. Visual Retriever Reader Pipeline.

Note: Given an image and a question, a visual retriever is first to retrieve relevant knowledge, and then a visual reader is to predict an answer based on the given knowledge.

visual information and evaluate a cross-modality model and a text-only caption-driven model (Section 4.1). On the reader side, we build two visual readers, a classification and an extraction type, with both utilizing visual information (Section 4.2). We observe in Chapter 6 that, our Visual-Retriever-Reader pipeline performs strongly on the OK-VQA challenge and establishes a new state-of-the-art.

1.2.3 Results Analysis

Listed as below, our experiments reveal several insights.

1. The image captions can largely promote the performance for both visual retriever and visual reader, which indicates the importance of applying image captioning generator to knowledge-based VQA tasks.
2. A neural retriever has much better performance than a term-based retriever. This observation is quite novel, as typically in the NLP domain, a term-based retriever (e.g., TF-IDF and BM25) is a hard-to-beat baseline (Lee, Chang, and Toutanova

2019; Lewis et al. 2020; Ma et al. 2020). Our results suggest an essential role of neural retrievers in the vision-&-language domain.

3. In the NLP domain, a reader can perform better if the given knowledge contains more relevant information to the question. Similarly, we discover that our visual reader has a significant leap between using noisy knowledge and high-quality knowledge. It motivates the demand for developing a more efficient visual retriever for knowledge-based VQA tasks.

In Chapter 7, we also analysis and evaluate the experiment results by using different predicting strategy (Section 4.3.2) and comparing to the complete corpus.

Chapter 2

RELATED WORK

2.1 Knowledge-based VQA

Many benchmarks have been proposed to facilitate the research in knowledge-based VQA. FVQA (P. Wang et al. 2017) is a fact-based VQA dataset that provides image-question-answer-supporting fact tuples, where the supporting fact is a structured triple, e.g., ⟨Cat, CapableOf,ClimbingTrees⟩. KB-VQA (P. Wang et al. 2015) dataset consists of three types of questions: "Visual" question answerable using the visual concept in an image, "Common-sense" questions answerable by adults without looking for an external source, and "KB-knowledge" questions requiring higher-level knowledge, explicit reasoning, and external resource. KVQA (Shah et al. 2019) consists of questions requiring world knowledge of named entities in images. Specifically, the questions require multi-entities, multi-relation, multi-hop reasoning over Wikidata. KVQA is challenging as linking the named entities in an image to the knowledge base is hard on a large scale. text-KVQA (Singh et al. 2019) focuses more on the texts shown in the image, which requires OCR technique to extract. In text-KVQA, the dataset is split in three parts: business, books and movies, and each parts requires external knowledge to the answer. For example, given a book cover, it requires knowledge to understand its content. OK-VQA (Marino et al. 2019) covers 11 types of knowledge than previous tasks, such as cooking and food, science and technology, plants and animals, etc. VLQA (Sampat, Yang, and Baral 2020) consists of data points of

image-passage-question-answer, it is proposed recently to facilitate the research on jointly reasoning with both image and text.

2.2 OK-VQA Models

Out of the Box (Narasimhan, Lazebnik, and Schwing 2018) utilizes the Graph Convolution Networks (Kipf and Welling 2016) to reason on the knowledge graph (KG), wherein each node image and semantic embeddings are attached. Mucko (Z. Zhu et al. 2020) goes a step further, reasoning on visual, fact, and semantic graphs separately, and uses cross-modal networks to aggregate them together.

As pretrained model BERT (Devlin et al. 2018) has achieved great success in a wide range of fields, it is also applied to language and vision cross-model recently, such as LXMERT (Tan and Bansal 2019), VLBERT (Su et al. 2019) and VilBERT (Lu et al. 2019). ConceptBert (Gardères et al. 2020) combines the BERT-pretrained model (Devlin et al. 2018) with KG. It encodes the KG using a transformer with a BERT embedding query. KRISP (Marino et al. 2020) involves a BERT-pretrained transformer model to make a better semantic understanding and utilize the implicit knowledge and reasons on a GCN model.

Recently some knowledge-oriented models are proposed to address OKVQA challenge. Span-Selector (Jain et al. 2021) extracts spans from the question to search most relative knowledge from Google, whereas MAVEx (Wu et al. 2021) votes among textual and visual knowledge from Wikipedia, ConceptNet, and Google Image. Besides knowledge collection, knowledge alignment (Shevchenko et al. 2021) also helps acquire a correct answer from knowledge.

2.3 Open-Domain Question Answering

Open-Domain Question Answering (ODQA) tasks target collecting information from a large corpus to answer a question. The advanced reading comprehension model (Chen et al. 2017) split this complex task into two steps: a retriever selects some most relevant documents from a corpus to a question, and a reader produces answer according to the documents from retriever. Some previous work (Kratzwald and Feuerriegel 2018; Lee et al. 2018; Das et al. 2019; S. Wang et al. 2018) train the end-to-end models to rerank in a closed set. Although these models are better at retrieval, they can hardly scale to larger corpora. Open-Retrieval Question Answering (ORQA) (Lee, Chang, and Toutanova 2019) and Dense Passage Retriever (DPR) (Karpukhin et al. 2020) constructed a dual-encoder architecture with BERT pre-trained model. This dense retrieval model shows a better performance than classic TF-IDF or BM25-based ODQA models.

Chapter 3

KNOWLEDGE CORPUS CREATION

The overall process of knowledge corpus creation (Figure 4) consists of following four steps.

3.1 Step 1: Query Preparation

Based on the assumption that the knowledge used for answering training set questions can also help in testing, the OK-VQA training questions are used with their answers to collect related knowledge from a search engine. We concatenate each question with each answer to get a "Question, Answer" pair. For example, in Figure 4, the question "What is the natural habitat of these animals?" has four answers, and each answer is attached to the question one by one to construct four queries.

3.2 Step 2: Google Search Webpage

The generated queries are sent to Google Search API to obtain knowledge. As presented in Figure 4, a good search result web page contains a title, a link, and a snippet that consists of multiple complete or incomplete sentences and shows the most relevant part to the query. The top *ten* web pages with their snippets as the raw knowledge are chosen.

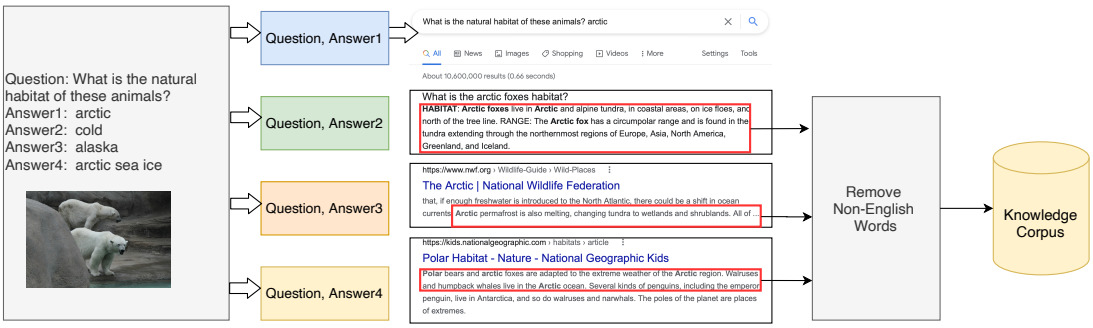

Figure 4. The overall process of Knowledge Corpus Creation.

Note: The question first combines the answers one by one to form a query, and then the query is sent to the Google Search API to retrieve the top *10* webpages. The knowledge is obtained from the snippet with further processing. Finally, we integrate the knowledge into the corpus. As shown in the searching result page, the black boxes represent webpages, and red boxes represent snippets.

3.3 Step 3: Snippet Processing

The snippets from Google searching results consist of multiple sentences, some are complete but some are not. One option is to split snippets into multiple sentences, but experimental result shows sentence-level knowledge is worse than snippet-level. Thus, we choose to use snippet as a knowledge. To address incomplete sentence issue, we find and grab the complete sentence present in the webpage. After this pre-processing, ten snippet-knowledge from each "Question, Answer" query are selected.

3.4 Step 4: Knowledge Processing

We first remove the duplicated data among each "Question, Answer" pair. Then long knowledge (more than 300 words) or short knowledge (less than ten words)

are removed. Pycld2[2] is applied in this step to detect and remove the non-English part of each knowledge. Each knowledge is assigned a unique ID and duplicate knowledge sentences are removed. We curate in total 112,724 knowledge sentences for the OK-VQA training set.

[2]https://pypi.org/project/pycld2/

Chapter 4

VISUAL RETRIEVER-READER PIPELINE

We present our Visual Retriever-Reader pipeline for the OK-VQA challenge, where the visual retriever aims to retrieve relevant knowledge, and the visual reader aims to predict answers given knowledge sentences. This scheme has been widely used in NLP (Chen et al. 2017; Karpukhin et al. 2020). While previous work focuses on pure text-domain, we extend this to the visual domain with novel adaptation.

4.1 Retriever

In this section, two styles of visual retriever were introduced: term-based and neural-network-based. In the neural style, we further introduce two variants. Following the convention, we use the standard terms in next subsection, for example, in Section 4.1.1, we use *documents* and in Section 4.1.2, we use *context*, both of them are *knowledge* in our task.

4.1.1 Term-based Retriever

BM25 is a widely-used algorithm in information retrieval (IR). In BM25 (Robertson and Zaragoza 2009), each query and document is represented by sparse vectors in d dimension space, where d is the vocabulary size. Then the score of a query and a document is computed based on the inverse term's frequency. BM25 can only retrieve documents for a query in text format, but an image is a part of a query in our task. To

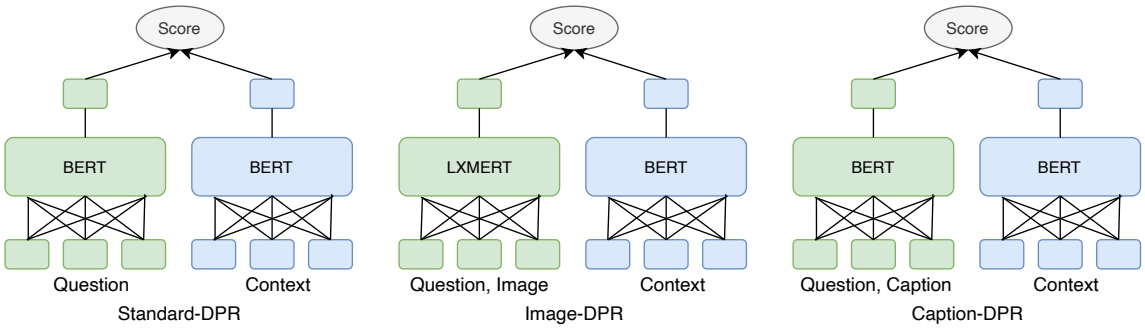

Figure 5. Comparison between standard DPR, Image-DPR and Caption-DPR.

Note: While the context encoder is the same for three models, in standard BERT(left), the question encoder only takes question as input, the Image-DPR(middle) takes both question and image as input, the Caption-DPR (right) takes the question and the caption as input.

tackle this issue, we first generate image captions using the latest caption generation model Oscar (Li et al. 2020). Then we concatenate the question and the caption as a query and obtain a list of documents by BM25.

4.1.2 Neural-based Retriever

Unlike BM25, neural retrievers extract the dense representations for a query and a context from the neural model(s). We use DPR (Karpukhin et al. 2020) as a neural retriever, which employs two BERT (Devlin et al. 2018) models to encode the query and context respectfully, then applies inner-dot product to estimate the relevancy between a query and a context. Similar to BM25, the DPR model considers the query in text format.To adapt DPR in the visual domain, we propose two methods: *Image-DPR* and *Caption-DPR*.

4.1.2.0.1 Image-DPR

In *Image-DPR*, we use LXMERT (Tan and Bansal 2019) as the question encoder, which takes image and question as input and outputs a cross-modal representation. For context encoder, we use the standard BERT. To train the retriever, we use inner-dot product function to get the similarity score of relevant and irrelevant knowledge to a question, and optimize the negative log likelihood of the relevant knowledge.

4.1.2.0.2 Caption-DPR

As its name suggests, we leverage the caption to capture the visual feature in *Caption-DPR*. Similar to the strategy we use in term-based retriever, we concatenate the question with the caption of an image as a query and use standard BERT as a query encoder to get the representation. Here the captions are also generated by the Oscar model. The resting parts remain the same as the *Image-DPR*. Figure 5 shows the architectures of standard DPR, *Image-DPR* and *Caption-DPR*.

4.1.3 Retrieval Results

Figure 6 shows the top knowledge retrieved by Term-based Retriever, Image-DPR and Caption-DPR. In this example, for the term-based retriever, key word "motorcycle" and "parking" show up several times in the retrieved knowledge, but the key word "sport" in the question is missing, resulting in the knowledge cannot answer the question. However, in both neural-network retriever, the results not only count into the effect of question, but also contain the correct answer, "race" or "racing".

Question: What sport can you use this for?
Caption: A black motorcycle parked in a parking lot.
Answers: race; motocross; ride

Term-Based Retriever
the sfmta designates a variety of parking spaces in metered areas, non-metered areas and off street lots and garages for motorcycle parking. the city of san francisco motorcycle parking map displays the locations of metered and non-metered on-street motorcycle parking.

Image-DPR
racing has been one of the most exciting forms of sports competition the human race has ever created. in this type of racing are from various classes, such as production cars, trucks and motorcycles, and it is usually recreational.

Caption-DPR
motorcycle racing is an electrifying sport, requiring a unique skillset and courageous dedication to the sport. the repsol honda team has shown superb performance in motogp, with over 100 triumphs and even three consecutive triple crown wins.

Figure 6. Examples of Knowledge Retrieved by 3 Retrievers.
Note: All the three knowledge are the top knowledge from the retriever results.

This example shows that the modified DPR models are able to retrieve the correct knowledge for OK-VQA questions.

4.2 Reader

In this section, two styles of readers are designed to predict an answer given the visual-linguistic features with the retrieved context: the classification reader (CReader) and the extraction reader (EReader).

4.2.1 Classification Reader

Current state-of-the-art VQA systems are classification models (Tan and Bansal 2019; Li et al. 2019; Gokhale et al. 2020), where a list of answer candidates are pre-defined (from the training set), i.e., a fixed answer vocabulary, then a model classifies one of the answers as the final prediction. We build a reader in this style but incorporate external knowledge. In particular, given a question, an image, and a piece of knowledge, we first concatenate the question with the knowledge and then apply a cross-modality model to encode the text with the image and generate a cross-modal representation. We feed this representation to a Multiple Layer Perceptron (MLP) which finally predicts one of the pre-defined answers. We apply Cross-Entropy Loss to optimize the model. In this work, we use LXMERT (Tan and Bansal 2019), while any other cross-modality models like VisualBERT (Li et al. 2019) can be adapted.

4.2.2 Extraction Reader

The classification model fails to generalize to out-of-domain answers, i.e., questions whose answers are not in the pre-defined answer vocabulary. To tackle this issue, we use an extraction model which is adapted from machine reading comprehension model (Chen et al. 2017). The model extracts a span (i.e., a start token and an end token) from the knowledge to answer the question. The image caption is given to the model as well to incorporate the image information. We also inject a special word "unanswerable" before the caption so that the model can predict "unanswerable" if the given knowledge can not be relied on to answer the question. This strategy is helpful since the retrieved knowledge might be noisy. We use a RoBERTa-large (Liu et al. 2019) as the text

encoder, whose inputs are {[SEP] question [SEP] ["unanswerable"], caption, knowledge [SEP]}. Then each token representation is fed to two linear layers: one predicts a score for a token being the start token, and the other predicts a score for the end token. We apply the softmax function to get the probability of each token being start and end token. The training objective is to maximize the probability of the ground truth start and end token.

4.2.3 Reader Outputs

Figure 7 shows one example answered by CReader and EReader. This example proves that the CReader correctly predict the answer several times, and we apply a predicting strategy (See Section 4.3.2) to select one answer from the multiple predictions.

EReader correctly answered this question, but the answer is not in the answer set provided. To address this issue, we introduced an Open-Domain Evaluation in Section 5.2.2.

4.3 Strategies

This section would introduce some strategies we used in the retrievers and readers.

4.3.1 Weak Supervision

We trained the retriever and the reader using weak supervision, where the ground-truth knowledge context is unknown for a given question-image pair.

Figure 7. Examples of Answer Predicted by 2 Readers.

Note: Here the CReader predicted 45 times motorcycles and 5 times motocross when providing total 50 knowledges. EReader predicted racing, which is a correct answer but not in the answer set.

For the retriever, given a query and an image, we assume that knowledge that contains any of the answers is relevant, and we use the in-batch negative samples (Karpukhin et al. 2020) for training, i.e., in the training time, any relevant knowledge for other questions in the same batch are considered as irrelevant. For the reader, we use the same relevant knowledge as the retriever; in addition, we use the collected knowledge from Google, which does not contain any answer as the irrelevant knowledge. If irrelevant knowledge is given, the reader should predict "unanswerable".

4.3.2 Prediction Strategy

We use the retriever to retrieve K knowledge (the value and effects of K will be presented in Section 7.2), and the reader predicts an answer based on each knowledge. We propose and compare the following two strategies to predict the final answer (See

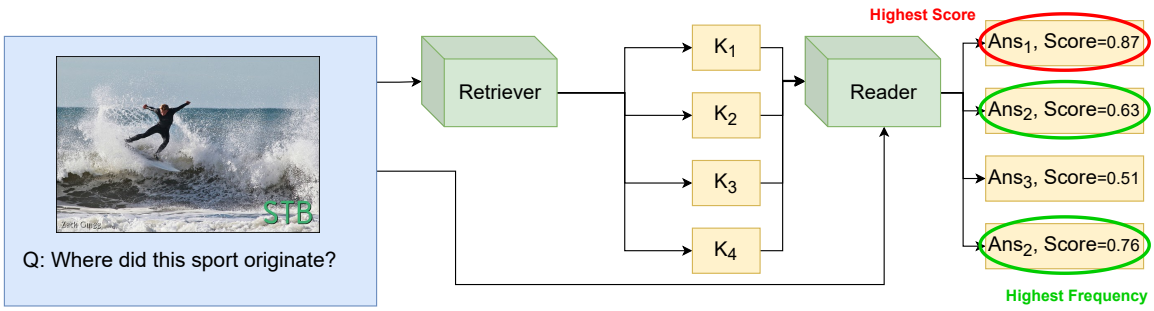

Figure 8. Highest-Score Strategy and Highest-Frequence Strategy

Note: In this example, 4 knowledges are retrieved to predict 4 answers, in which Ans_1 scores the highest, and Ans_2 appears twice. We pick Ans_1 as the Highest-Score prediction, and Ans_2 as the Highest-Frequency prediction.

Figure 8). *Highest-Score*: the answer which has the highest score is the final prediction.

Highest-Frequency: the answer which appears most frequently is the final prediction.

Chapter 5

EVALUATION

In this chapter, we introduce our evaluation method for the retrievers and readers. The retrievers are evaluated using metrics like Precision, Recall and F1 score, which are commonly used in the NLP fields. For the readers, we adopt the general VQA evaluation method. In order to avoid the effect of outer words bring from the Reading Comprehension Model by the greatest extent, we also use the entailment method (Luo et al. 2021) to re-evaluate the reader predictions (see Section 5.2.2), which largely improved our SOTA accuracy to 45.8%.

5.1 Retriever Evaluation

We evaluate the performance of a retriever based on Precision, Recall and F1 score. The two metrics are based on the assumption that any retrieved knowledge that contains any of the answers annotated in the OK-VQA dataset is relevant. This assumption is because it is unknown which knowledge is relevant to a question-image pair. Therefore the computation of Precision and Recall in our case is different from the traditional definition and illustrated as follow:

5.1.1 Precision

Precision reveals the proportion of retrieved knowledge that contains any of the answers to a question-image pair. Mean Precision is the mean of Precision of all

question-image pairs. Mathematically,

$$P(Q, A, KN) = \frac{1}{K} \sum_{i=1}^{i=K} min(\sum_{\substack{j=1 \\ A_j \in KN_i}}^{j=M} 1, 1),$$

where Q is a question, KN is a list of retrieved knowledge, A is a list of correct answers, K is the number of KN, M is the number of A.

5.1.2 Recall

Recall reveals if at least one knowledge sentence in the retrieved Knowledge contains any answers to a question-image pair. Mean Recall is the mean of the Recall of all question-image pairs. Mathematically,

$$R(Q, A, KN) = min(\sum_{i=1}^{i=K} \sum_{\substack{j=1 \\ A_j \in KN_i}}^{j=M} 1, 1),$$

where the meaning of the symbols are the same described in Precision.

5.1.3 F1 Score

F1 score takes both the precision and the recall into account, and can be considered as the harmonic mean of the precision and the recall. Mathematically,

$$F1(Q, A, KN) = 2 \cdot \frac{P(Q, A, KN) \cdot R(Q, A, KN)}{P(Q, A, KN) + R(Q, A, KN)}$$

5.2 Answer Evaluation

5.2.1 Standard VQA Evaluation

In OK-VQA, each image-question pair has five answers annotated by humans. To apply a similar evaluation as VQA (Agrawal et al. 2015), OK-VQA counts per answer twice so that each image-question pair has ten answers, the same as VQA. The score is computed as follows.

$$score(A) = min(\frac{\#\text{human that said A}}{3}, 1)$$

We use the above equation to compute the score of each answer for training and testing.

5.2.2 Open-Domain Evaluation

Considering that the Extraction Reader predicts an answer within the open domain, probably resulting in the generated phrases not showing up in the answer field, we introduced a novel open-domain evaluation using Sentence Textual Entailment (STE) tool. This evaluating work contains two phases: *Grounding* that apply each answer and prediction to the question to ground it as a statement; and *Entailment* that calculate the similarity of the different grounded sentences. Then the final score is calculated according to the STE results. One example is shown in Figure 9.

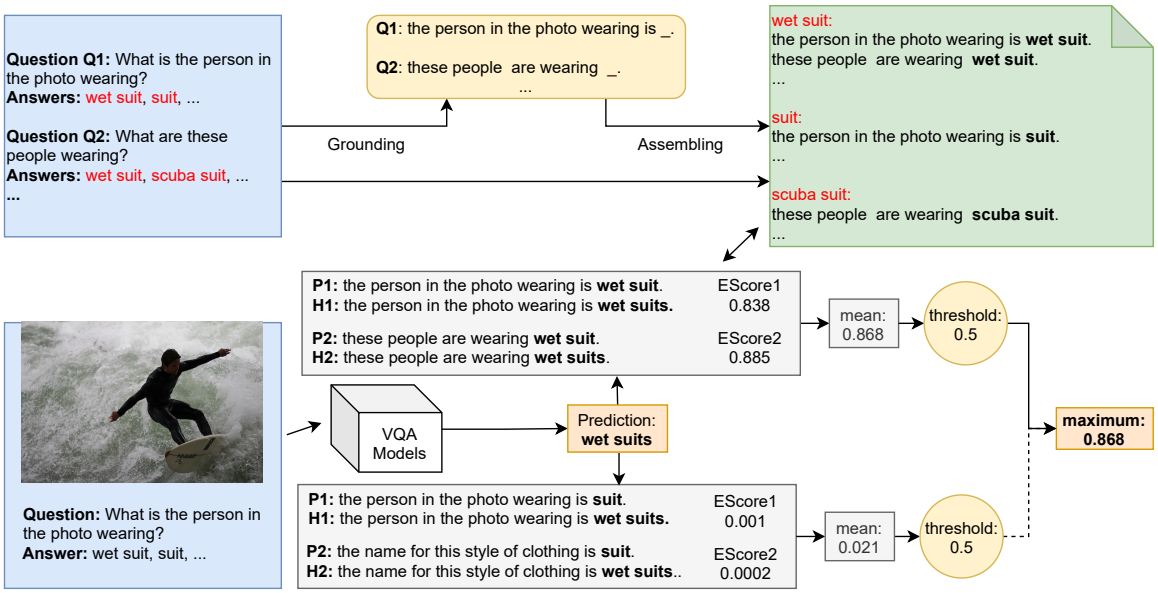

Figure 9. Example of Open-Domain Evaluation

Note: This example calculates the entailment score of provided answer "wet suit" and our prediction "wet suits". We first ground all questions into statements with a reserved position "_" for the answer. Then, we congregate all the grounded statements by the provided answer. We replace the "_" with the provided and predicted answer separately as the premise and hypothesis to get the entailment score. The entailment score of a provided answer and a prediction is calculated as the mean of all the entailment scores under that answer in the assembling list. We take 0.5 as the threshold, and use the maximum as the final entailment score.

5.2.2.1 Grounding

In the grounding phase, we convert a question to a statement using the answers and predictions. Since a good prediction should be of the similar semantic meaning as the answers, we assume that for one question, every answer and prediction acts as the same role in the grounded statement, and thus we ground the question with a reserved position for any answer to fill in. For example, the original question "Who invented this device?" is grounded to "_ invented this device.", where "_" can be any of the answers to this question. An example for grounding is shown in Figure 10.

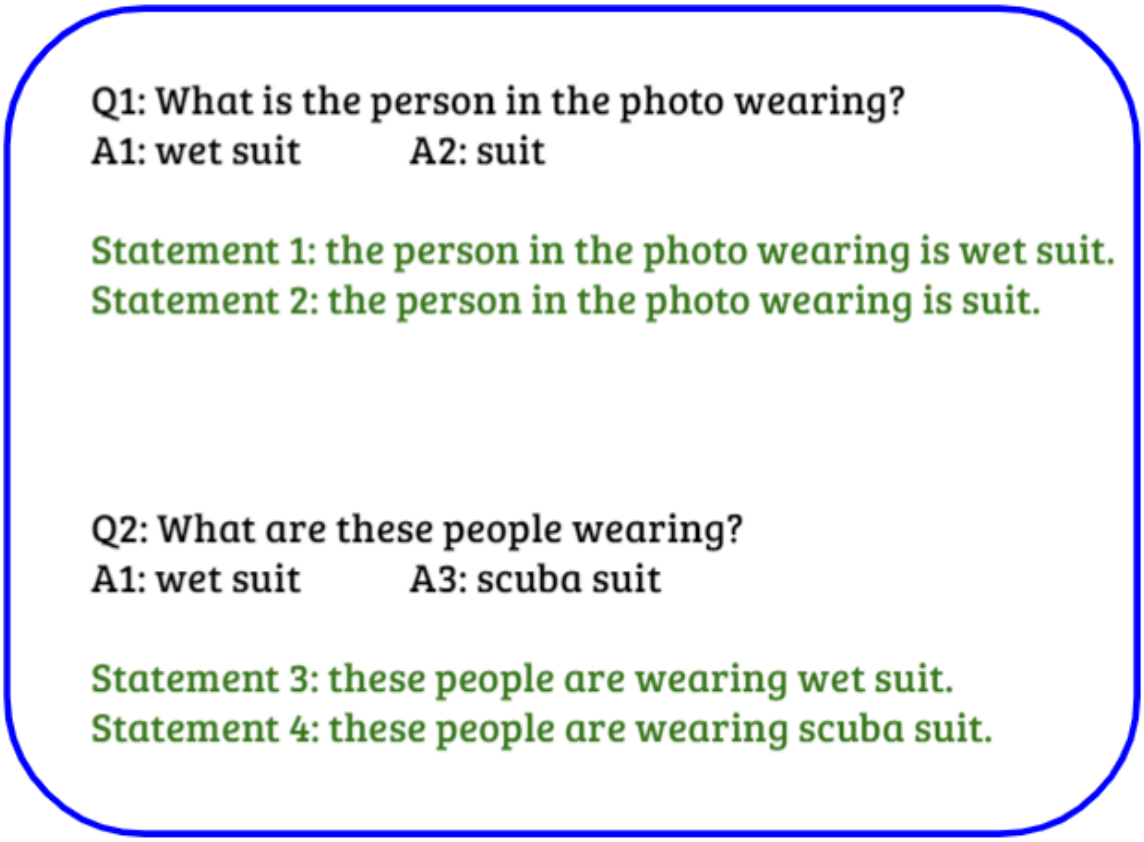

Figure 10. Example of Grounding Step in Open-Domain Evaluation

To achieve this, a simple sentence role labeling work is applied to the questions to detect different elements in the sentence (question word, object, subject, auxiliary word, etc.). After settling the role of elements, the question is then re-ordered to accord with the word order of declarative sentences.

We apply the above method to the wh-questions and choice questions, which in total cover the 98.6% of questions and 98.9% of unique answers. Table 2 shows some examples of grounded sentences.

Table 2. Examples for some grounded sentences where the hypothesis gets score over the threshold.

Original Question	Grounded Statement
What is this type of blanket called?	this type of blanket is called _.
What is the name of the board he is on?	the name of the board he is on is _.
The food in the photo contains which healthy vitamins?	The food in the photo contains _ healthy vitamins.
Is this bathroom high or low end?	this bathroom is _.
Why is the cow going to the water?	the cow is going to the water because of _.

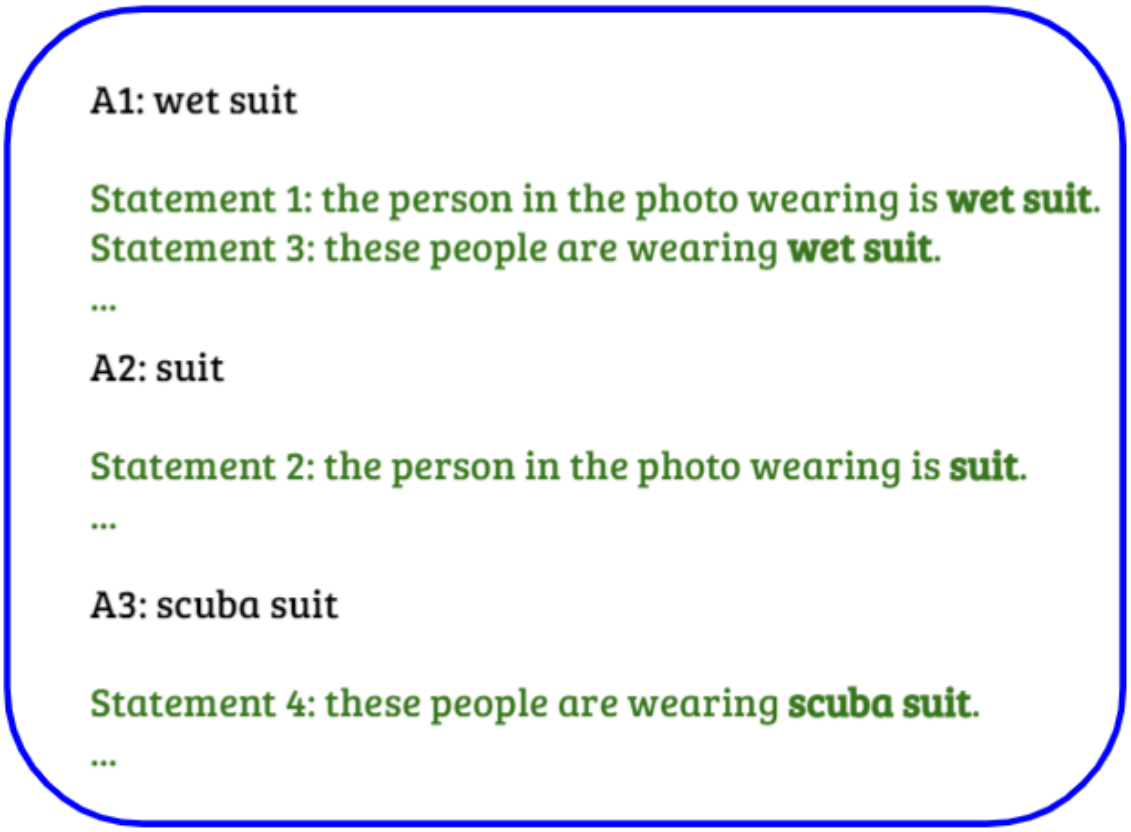

Figure 11. Example of Assembling Step in Open-Domain Evaluation

5.2.2.2 Assembling

In grounding step, the statements are gathered by question. We re-arrange the these grounded statements ordered by the provided answers for the further processing. Figure 11 provides an example for this assembling step.

5.2.2.3 Entailment

The grounded sentences are then sent to the Natural Language Inference (NLI) model [3]. NLI is used widely in the NLP tasks to check whether the hypothesis can be entailed from the given premise, and here we use NLI to check whether the provided answers and the predicted answer are semantically the same. To compare between a provided answer and a predicted answer, we first list all grounded statements that use the provided answer as a correct answer. Then, for each of these statements, we fill the reserved position with the provided answer as the premise, and our prediction is the hypothesis, and calculate the entailment score. We use the arithmetic mean of these scores as the final entailment score.

The threshold is set to be 0.5. We also skip the choice questions and the questions with numbers as answers, since, with only grounded statements provided, it is hard to tell whether the two numbers or two choices are similar. For each question with multiple answers, we pick the highest entailment score as the similarity score.

5.2.2.4 Result

Finally we use the following equation to calculate the re-evaluated accuracy:

$$S(A) = argmax(\sum_{g_i \in G_{Ans}} E(A, Ans, g_i))) \cdot max(\frac{1}{N} \cdot \sum_{g_i \in G_{Ans}} E(A, Ans, g_i))),$$

where the E(A, Ans, g_i) denotes the entailment score given a prediction A, a correct answer Ans and a grounded sentence g_i, and argmax() picks the original score of

[3]https://github.com/allenai/allennlp-models/tree/v1.0.0.rc2/training_config/nli

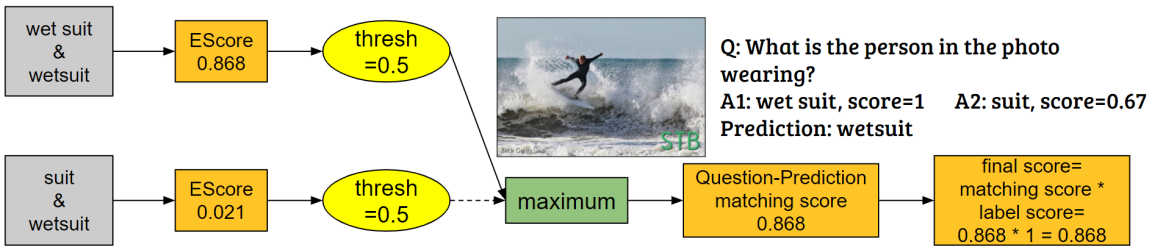

Figure 12. Example of Entailment Step in Open-Domain Evaluation

the correct answer corresponding to the highest NLI score. Here G_{Ans} is the set of grounded sentences that uses Ans as answer, and N denotes the size of G_{Ans}.

Figure 12 shows the steps acquiring the entailment score and calculating the final score for a predicted answer.

This open-domain evaluation promotes our SOTA model up to 47.3% score.

Chapter 6

EXPERIMENTS AND RESULTS

6.1 Baselines

We use a state-of-the-art vision-language model, LXMERT (Tan and Bansal 2019), as the baselines and apply Captioning and Optical Character Recognition (OCR) results to the OK-VQA dataset to the original LXMERT model.

6.1.1 LXMERT

LXMERT is a BERT-based cross-modality model pretrained on five different VQA datasets: MS COCO (Lin et al. 2014), Visual Genome (Krishna et al. 2017), VQA v2.0 (Antol et al. 2015), GQA balanced version (Hudson and Manning 2019) and VG-QA (Y. Zhu et al. 2016). We fine-tune LXMERT on OK-VQA and surprisingly find that LXMERT ranks higher than most of the SOTA models, for which reason we set LXMERT as our baseline model.

6.1.2 LXMERT with OCR

The OCR technique captures the textual contents from the image and transfers them into characters. Here we use Google Vision API[4] to extract the texts from images. After the noise deduction step filtering all non-English words, we attach the

[4]https://cloud.google.com/vision/

OCR results after the question and then sent them into the LXMERT model. Our experiment shows that the OCR result helps to address the OK-VQA task.

6.1.3 LXMERT with Captioning

Similar to OCR, we also experiment with adding captioning when training the LXMERT model. The captions are generated by the advanced model Oscar (Li et al. 2020) and attached to each question when sent into the LXMERT model. Our result shows that captioning improves the performance of the LXMERT model, and therefore, we put the LXMERT with captioning as a baseline as well.

6.2 Main Results

We performed all the experiments at GTX1080 and V100 NVIDIA GPUs. For both Image-DPR and Caption-DPR, we set the training epoch to be 30, learning rate (lr) be 1e-5, batch size (bs) be 8, gradient accumulation step (gas) be 4. In CReader, we set the training epoch as 3, lr as 2e-5, and batch-size as 16. In EReader, we set the training epoch as 3, lr as 1e-5, batch-size as 4, and gradient accumulation as 4.

6.2.1 Performance

Table 3 shows that our best model based on Caption-DPR and EReader outperforms previous methods and establishes the new state-of-the-art result on the OK-VQA challenge. Interestingly, the LXMERT baseline without utilizing any knowledge

Table 3. Performance on the OK-VQA Test-split.

Method	**Knowledge Src.**	**Acc.**
Existing Method		
KRISP (Marino et al. 2020)	W & C	32.3
ConceptBert (Gardères et al. 2020)	C	33.7
MAVEx (Wu et al. 2021)	W & C & GI	38.7
Baselines		
LXMERT	-	36.2
LXMERT + OCR	-	37.2
LXMERT + Caption	-	37.8
LXMERT + OCR + Caption	-	37.2
Visual Retriever-Reader		
BM25 + CReader	GS	35.13
BM25 + EReader	GS	32.10
Image-DPR + CReader	GS	34.64
Image-DPR + EReader	GS	33.95
Caption-DPR + CReader	GS	36.78
Caption-DPR + EReader	GS	**39.20**
Caption-DPR + EReader †	GS	59.22

Note: Our model outperforms existing methods. † means given oracle knowledge to the reader. GS-Google Search (Training Corpus). W-Wikipedia, C-ConceptNet, GI-Google Image, Acc-Accuracy.

Table 4. Evaluation of three proposed visual retrievers on Precision, Recall and F1 score

Model	**# of Retrieved Knowledge**														
	1			5			10			50			100		
	P*	R*	F1	P*	R*	F1	P*	R*	F1	P*	R*	F1	P*	R*	F1
BM25	37.63	37.63	37.63	35.21	56.72	43.45	34.03	67.02	45.14	29.99	84.56	44.27	27.69	89.91	42.34
Image-DPR	33.04	33.04	33.04	31.80	62.52	42.16	31.09	73.96	43.78	28.55	90.84	43.44	26.75	94.67	41.71
Caption-DPR	**41.62**	**41.62**	**41.62**	**39.42**	**71.52**	**50.83**	**37.94**	**81.51**	**51.78**	**32.94**	**94.13**	**48.80**	**30.01**	**96.95**	**45.83**

Note: Caption-DPR achieves the highest Precision, Recall and F1 Score on all number of retrieved knowledge. We have a * marker on the **P**recision and **R**ecall to distinguish from traditional Precision and Recall as illustrated in Section 5.1.

achieves better performance than KRISP (Marino et al. 2020) and ConceptBert (Gardères et al. 2020) which leverage external knowledge. Incorporating OCR and captioning further improve the baseline accuracy by 1% and 1.6%, respectively.

Among different variations of Visual Retriever-Reader, the best combination is Caption-DPR and CReader when the retrieved knowledge size is 80. We evaluate retrievers' performance based on Precision, Recall and F1 score. Table 4 shows that Caption-DPR consistently outperforms BM25 and Caption-DPR on the various number of retrieved knowledge. It is interesting to see that Caption-DPR outperforms BM25 significantly since BM25 is a hard-to-beat baseline in open-domain QA (Lee, Chang, and Toutanova 2019; Lewis et al. 2020; Ma et al. 2020). It indicates that neural retriever has better application than term-based retrieval methods in the vision domain.

Table 5. Recall increases when the Caption-DPR method retrieves knowledge from a complete knowledge corpus created using train and test questions.

Model	**# of Retrieved Knowledge**						
	1	5	10	20	50	80	100
BM25	+6.00	+6.28	+4.88	+4.32	+3.83	+3.17	+2.56
Image-DPR	+2.24	+2.60	+2.93	+2.29	+1.83	+1.29	+1.25
Caption-DPR	+8.88	+8.88	+7.04	+4.65	+2.98	+2.23	+1.88

Chapter 7

ANALYSIS AND DISCUSSION

Based on the experiments in Chapter 6, we do some further analysis that help explore the deeper essence of our Retriever-Reader Pipeline. We compare our SOTA model to the oracle one, which uses only the relevant knowledge to generate answer. We also evaluate the effect of different prediction strategies and the effect of the completeness of the corpus.

The detailed prediction may also review some essential features of our model. We are planing to compare the different predictions by the two styles of readers, and see where can our model be improved. This prediction analysis will be our future directions (Section 8.2.2).

7.1 Effects of the Quality of Knowledge.

A common observation in open-domain question answering in NLP is that the reader can perform well if the given knowledge is good to answer a question. Here, we are interested to see if this also holds for our reader. Specifically, we set the oracle-knowledge model as removing knowledge that does not contain any answer before we feed the retrieved knowledge to the reader, and sending the remaining knowledge to the reader. The last row in Table 3 shows that our reader can perform much better if the quality of the knowledge is good, suggesting that a more efficient cross-modality retriever is needed.

7.2 Effects of Size of Retrieving Knowledge and Prediction Strategy.

The performance of reader is directly affected by the size of retrieved knowledge. A more extensive knowledge set is more likely to include the relevant knowledge to answer the question yet along with more distracting knowledge. In contrast, a small set might exclude relevant knowledge but with fewer distracting knowledge. We use Caption-DPR to retrieve the different number of pieces of knowledge and use the EReader to predict an answer given the different number of pieces of knowledge. We compare the effects on two prediction strategies mentioned in Section 4.3.2. Figure 13 shows the comparison, and we have the following observations. First, when the knowledge size is small (equal or less than 5), the Highest-Score strategy is better than the Highest-Frequency; on the other hand, when the knowledge size is large, the Highest-Frequency strategy performs better than the Highest-Score strategy. Second, for the Highest-Score strategy, the size of 5 is the best, and increasing the knowledge size reduces the performance. Third, for the Highest-Frequency strategy, when the size equal to 80, it yields the best performance. To summarize, if one uses a small set of knowledge, then Highest-Frequency negatively impacts the accuracy and the Highest-Score strategy is preferable. If one uses a larger corpus of knowledge, the Highest-Frequency strategy can achieve higher accuracy.

7.3 Effects of Completeness of Corpus.

So far, when we test the model performance, we use the knowledge corpus collected only by training questions. However, if the entire training corpus does not include relevant knowledge to testing questions, our model is under-evaluated because of the

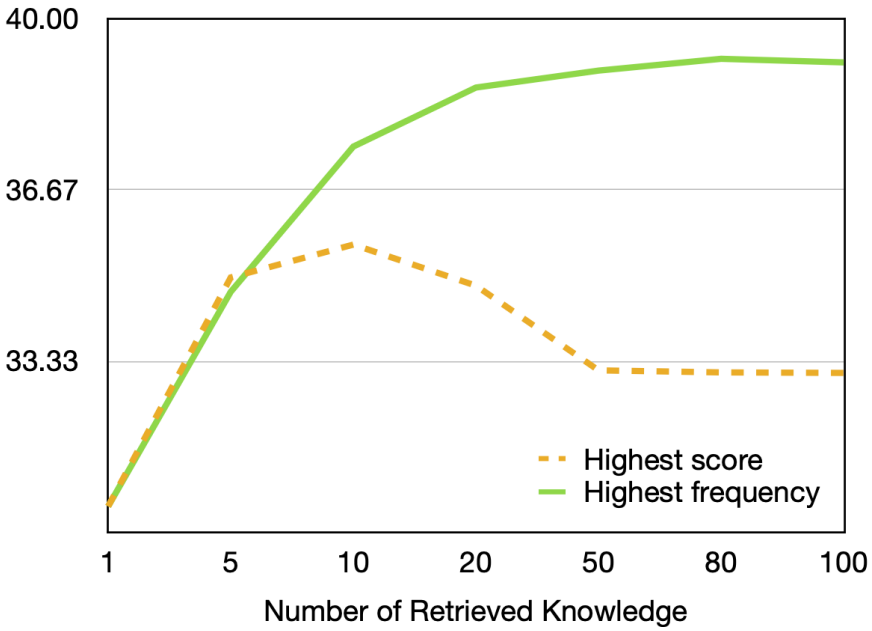

Figure 13. Highest-Score Strategy and Highest-Frequencey Strategy.

Note: Highest-Score Strategy: Performance of EReader decreases when the knowledge number increase and the best is at 5. Highest-Frequency Strategy: Performance of EReader increase when the knowledge number increase and the best is at 80.

incompleteness of the knowledge corpus. To fairly see how our model performs when the knowledge corpus is complete, we use the same knowledge collection method described in Section 3 to collect knowledge for testing questions. Then we combine the training and testing knowledge as a complete corpus, which increase the corpus size from 112,724 to 168,306. We use Caption-DPR to retrieve knowledge from the complete corpus and ask EReader to predict answers based on these pieces of knowledge. Table 5 shows the increase of recall. As we expected, a complete corpus is helpful for Caption-DPR even though the corpus size increased, thus yields better performance of EReader. Figure 14 compares the accuracy of EReader using knowledge retrieved from two corpora. EReader consistently achieves higher performance using

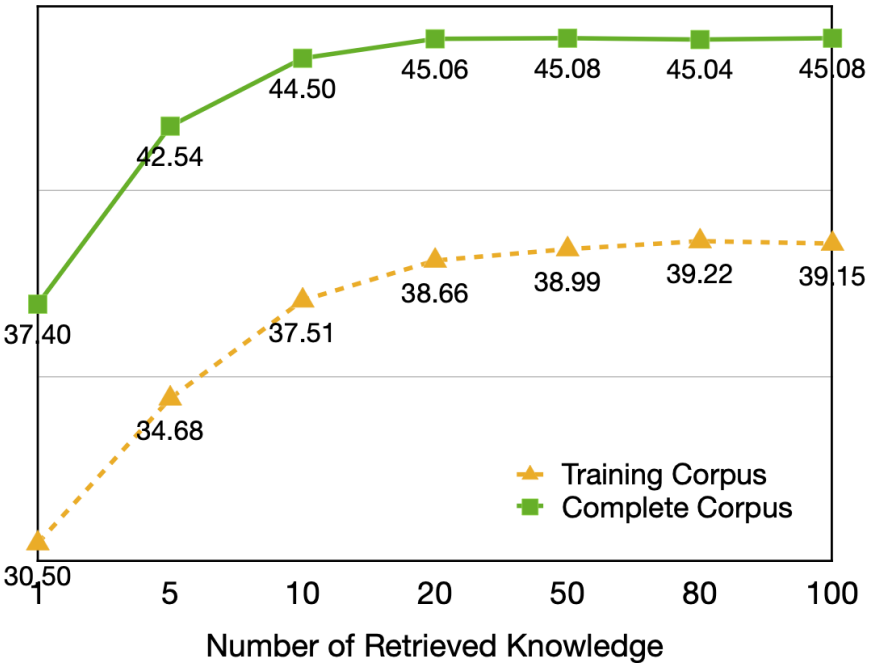

Figure 14. EReader achieves significant improvement when using knowledge retrieved from complete corpus compared to knowledge from training corpus.

the knowledge retrieved from complete corpus, where the biggest gain of 7.86% is achieved when using 5 knowledge.

7.4 Discussion

Although our pipeline is evaluated on the OK-VQA benchmark, it is generic and can be adapted for other knowledge-based question answering tasks such as FVQA (P. Wang et al. 2017), KB-VQA (P. Wang et al. 2015), KVQA (Shah et al. 2019), and text-KVQA (Singh et al. 2019). For example, in KVQA, we can first collect a named-entity knowledge corpus by the proposed knowledge collection approach and then apply our Visual-Retriever-Reader pipeline. It should be noted that our proposed

extraction reader is a more challenging problem as classification models tend to learn correlation between output classes (answers) (Agarwal, Shetty, and Fritz 2020) and input image and question. In contrast, the extraction reader extracts answer-spans which we exactly match with targets (answers).

Chapter 8

CONCLUSION AND FUTURE DIRECTIONS

8.1 Conclusion

In this work, we collect an easy-to-use free-form natural language knowledge corpus for VQA tasks with external knowledge. The corpus is collected on the training split of OK-VQA task by searching each "Question, Answer" pair through the Google Search API. We take the top ten results for each search, and do some further processing and finally get 112,724 knowledge.

We also construct a weakly-supervised Visual Retriever-Reader Pipeline, where the retriever consists of term-based BM25 model and neural-network-based Image-DPR and Caption-DPR, and the reader contains classification and extraction two styles. The Visual Retriever-Reader Pipeline has been evaluated on the OK-VQA challenge benchmark and has established a new state-of-the-art performance.

We set the baseline using LXMERT model with captioning and OCR, and the performance reveals that the captioning and the neural retriever can both significantly improve the QA system's performance. The further analysis, especially using oracle knowledge retrieved and using complete knowledge searched by testing set questions shows that good knowledge from the retriever makes vital progress in predicting the correct answer.

8.2 Future Directions

Although our model sets the SOTA for OK-VQA task, it still does not even reach 40% accuracy. As the experiments reveal that the knowledge is essential for the readers to predict a correct answer, and that there is a large leap between our best performance and the oracle score, how to retrieve better knowledge in retriever comes to a vital stage.

8.2.1 Knowledge Selection

As the oracle-knowledge model shown in the last row of the Table 3, the existing retriever is capable to retrieve the relevant knowledge, but there is some noise mixed into it. Therefore, we proposed a knowledge selecting model as the future work. This knowledge selecting model picks among the retrieved knowledge set to find the knowledge that is more possible to answer the question.

We consider the weakly-supervised model can hardly give a more specific ranking among the retrieved knowledge, and thus we manually annotated a subset of the knowledge base to precisely find which knowledge is right for answering the question. We first pre-train the knowledge selecting model using weakly-supervised method, then use this annotated subset to fine-tune the model. After the knowledge selection work, the most relevant knowledge may appear in a higher score among the retrieved knowledge.

8.2.2 Prediction Analysis

In Chapter 7 we analyze and evaluate the effect of different knowledge and predicting strategies on the Retriever-Reader Pipeline, but the prediction itself remains unanalyzed. However, the prediction worth analyzing and will probably expose some internal problems to us.

In future, we will look carefully into the predictions both by CReader and EReader, comparing their prediction with the baselines to see which categories our model does better than the LXMERT, and which categories not. We will also compare the different predictions under the same question but given different knowledge to see how the knowledge affect the answer prediction.

REFERENCES

Agarwal, Vedika, Rakshith Shetty, and Mario Fritz. 2020. "Towards causal vqa: Revealing and reducing spurious correlations by invariant and covariant semantic editing." In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition,* 9690–9698.

Agrawal, Aishwarya, Jiasen Lu, Stanislaw Antol, M. Mitchell, C. L. Zitnick, D. Parikh, and Dhruv Batra. 2015. "VQA: Visual Question Answering." *International Journal of Computer Vision* 123:4–31.

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

Chen, Danqi, Adam Fisch, Jason Weston, and Antoine Bordes. 2017. "Reading wikipedia to answer open-domain questions." *arXiv preprint arXiv:1704.00051.*

Das, Rajarshi, Shehzaad Dhuliawala, Manzil Zaheer, and Andrew McCallum. 2019. "Multi-step retriever-reader interaction for scalable open-domain question answering." *arXiv preprint arXiv:1905.05733.*

Devlin, Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. "Bert: Pre-training of deep bidirectional transformers for language understanding." *arXiv preprint arXiv:1810.04805.*

Gardères, François, Maryam Ziaeefard, Baptiste Abeloos, and Freddy Lecue. 2020. "Conceptbert: Concept-aware representation for visual question answering." In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings,* 489–498.

Gokhale, Tejas, Pratyay Banerjee, Chitta Baral, and Yezhou Yang. 2020. "VQA-LOL: Visual Question Answering under the Lens of Logic." *ArXiv* abs/2002.08325.

Hudson, Drew A, and Christopher D Manning. 2019. "Gqa: A new dataset for real-world visual reasoning and compositional question answering." In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition,* 6700–6709.

Iwana, Brian Kenji, Syed Tahseen Raza Rizvi, Sheraz Ahmed, Andreas Dengel, and Seiichi Uchida. 2016. "Judging a book by its cover." *arXiv preprint arXiv:1610.09204.*

Jain, Aman, Mayank Kothyari, Vishwajeet Kumar, Preethi Jyothi, Ganesh Ramakrishnan, and Soumen Chakrabarti. 2021. "Select, Substitute, Search: A New Benchmark for Knowledge-Augmented Visual Question Answering." *arXiv preprint arXiv:2103.05568.*

Karpukhin, Vladimir, Barlas Oğuz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. "Dense passage retrieval for open-domain question answering." *arXiv preprint arXiv:2004.04906.*

Kipf, Thomas N, and Max Welling. 2016. "Semi-supervised classification with graph convolutional networks." *arXiv preprint arXiv:1609.02907.*

Kratzwald, Bernhard, and Stefan Feuerriegel. 2018. "Adaptive document retrieval for deep question answering." *arXiv preprint arXiv:1808.06528.*

Krishna, Ranjay, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. 2017. "Visual genome: Connecting language and vision using crowdsourced dense image annotations." *International journal of computer vision* 123 (1): 32–73.

Lee, Jinhyuk, Seongjun Yun, Hyunjae Kim, Miyoung Ko, and Jaewoo Kang. 2018. "Ranking paragraphs for improving answer recall in open-domain question answering." *arXiv preprint arXiv:1810.00494.*

Lee, Kenton, Ming-Wei Chang, and Kristina Toutanova. 2019. "Latent retrieval for weakly supervised open domain question answering." *arXiv preprint arXiv:1906.00300.*

Lewis, Patrick, Ethan Perez, Aleksandara Piktus, Fabio Petroni, V. Karpukhin, Naman Goyal, Heinrich Kuttler, et al. 2020. "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks." *ArXiv* abs/2005.11401.

Li, Liunian Harold, Mark Yatskar, Da Yin, Cho-Jui Hsieh, and Kai-Wei Chang. 2019. "VisualBERT: A Simple and Performant Baseline for Vision and Language." *ArXiv* abs/1908.03557.

Li, Xiujun, Xi Yin, Chunyuan Li, Pengchuan Zhang, Xiaowei Hu, Lei Zhang, Lijuan Wang, Houdong Hu, Li Dong, Furu Wei, et al. 2020. "Oscar: Object-semantics aligned pre-training for vision-language tasks." In *European Conference on Computer Vision,* 121–137. Springer.

Lin, Tsung-Yi, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. 2014. "Microsoft coco: Common objects in context." In *European conference on computer vision,* 740–755. Springer.

Liu, Yinhan, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. "Roberta: A robustly optimized bert pretraining approach." *arXiv preprint arXiv:1907.11692.*

Lu, Jiasen, Dhruv Batra, Devi Parikh, and Stefan Lee. 2019. "Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks." *arXiv preprint arXiv:1908.02265.*

Luo, Man, Shailaja Keyur Sampat, Riley Tallman, Yankai Zeng, Manuha Vancha, Akarshan Sajja, and Chitta Baral. 2021. "'Just because you are right, doesn't mean I am wrong': Overcoming a bottleneck in development and evaluation of Open-Ended VQA tasks." In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume,* 2766–2771.

Ma, Ji, I. Korotkov, Yin-Fei Yang, K. Hall, and Ryan T. McDonald. 2020. "Zero-shot Neural Retrieval via Domain-targeted Synthetic Query Generation." *ArXiv* abs/2004.14503.

Marino, Kenneth, Xinlei Chen, Devi Parikh, Abhinav Gupta, and Marcus Rohrbach. 2020. "KRISP: Integrating Implicit and Symbolic Knowledge for Open-Domain Knowledge-Based VQA." *arXiv preprint arXiv:2012.11014.*

Marino, Kenneth, Mohammad Rastegari, Ali Farhadi, and Roozbeh Mottaghi. 2019. "Ok-vqa: A visual question answering benchmark requiring external knowledge." In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition,* 3195–3204.

Miller, George A. 1995. "WordNet: a lexical database for English." *Communications of the ACM* 38 (11): 39–41.

Narasimhan, Medhini, Svetlana Lazebnik, and Alexander G Schwing. 2018. "Out of the box: Reasoning with graph convolution nets for factual visual question answering." *arXiv preprint arXiv:1811.00538.*

Robertson, S., and H. Zaragoza. 2009. "The Probabilistic Relevance Framework: BM25 and Beyond." *Found. Trends Inf. Retr.* 3:333–389.

Sampat, Shailaja Keyur, Yezhou Yang, and Chitta Baral. 2020. "Visuo-Lingustic Question Answering (VLQA) Challenge." In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings,* 4606–4616.

Shah, Sanket, Anand Mishra, Naganand Yadati, and Partha Pratim Talukdar. 2019. "Kvqa: Knowledge-aware visual question answering." In *Proceedings of the AAAI Conference on Artificial Intelligence,* 33:8876–8884. 01.

Shevchenko, Violetta, Damien Teney, Anthony Dick, and Anton van den Hengel. 2021. "Reasoning over Vision and Language: Exploring the Benefits of Supplemental Knowledge." *arXiv preprint arXiv:2101.06013.*

Singh, Ajeet Kumar, Anand Mishra, Shashank Shekhar, and Anirban Chakraborty. 2019. "From strings to things: Knowledge-enabled VQA model that can read and reason." In *Proceedings of the IEEE/CVF International Conference on Computer Vision,* 4602–4612.

Speer, Robyn, Joshua Chin, and Catherine Havasi. 2017. "Conceptnet 5.5: An open multilingual graph of general knowledge." In *Proceedings of the AAAI Conference on Artificial Intelligence,* vol. 31. 1.

Su, Weijie, Xizhou Zhu, Yue Cao, Bin Li, Lewei Lu, Furu Wei, and Jifeng Dai. 2019. "Vl-bert: Pre-training of generic visual-linguistic representations." *arXiv preprint arXiv:1908.08530.*

Tan, Hao Hao, and Mohit Bansal. 2019. "LXMERT: Learning Cross-Modality Encoder Representations from Transformers." In *EMNLP/IJCNLP.*

Vrandečić, Denny, and Markus Krötzsch. 2014. "Wikidata: a free collaborative knowledgebase." *Communications of the ACM* 57 (10): 78–85.

Wang, Peng, Qi Wu, Chunhua Shen, Anthony Dick, and Anton Van Den Hengel. 2017. "Fvqa: Fact-based visual question answering." *IEEE transactions on pattern analysis and machine intelligence* 40 (10): 2413–2427.

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

Wang, Shuohang, Mo Yu, Xiaoxiao Guo, Zhiguo Wang, Tim Klinger, Wei Zhang, Shiyu Chang, Gerry Tesauro, Bowen Zhou, and Jing Jiang. 2018. "R 3: Reinforced ranker-reader for open-domain question answering." In *Proceedings of the AAAI Conference on Artificial Intelligence,* vol. 32. 1.

Wu, Jialin, Jiasen Lu, Ashish Sabharwal, and Roozbeh Mottaghi. 2021. "Multi-Modal Answer Validation for Knowledge-Based VQA." *arXiv preprint arXiv:2103.12248.*

Zhu, Yuke, Oliver Groth, Michael Bernstein, and Li Fei-Fei. 2016. "Visual7w: Grounded question answering in images." In *Proceedings of the IEEE conference on computer vision and pattern recognition,* 4995–5004.

Zhu, Zihao, Jing Yu, Yujing Wang, Yajing Sun, Yue Hu, and Qi Wu. 2020. "Mucko: Multi-Layer Cross-Modal Knowledge Reasoning for Fact-based VisualQuestion Answering." *arXiv preprint arXiv:2006.09073.*