Passage Retrieval for Outside-Knowledge Visual Question Answering

Chen Qu, Hamed Zamani, Liu Yang, W. Bruce Croft, Erik Learned-Miller
University of Massachusetts Amherst
Amherst, MA, United States
{chenqu,zamani,lyang,croft,elm}@cs.umass.edu

ABSTRACT

In this work, we address multi-modal information needs that contain text questions and images by focusing on passage retrieval for outside-knowledge visual question answering. This task requires access to outside knowledge, which in our case we define to be a large unstructured passage collection. We first conduct sparse retrieval with BM25 and study expanding the question with object names and image captions. We verify that visual clues play an important role and captions tend to be more informative than object names in sparse retrieval. We then construct a dual-encoder dense retriever, with the query encoder being LXMERT [35], a multi-modal pre-trained transformer. We further show that dense retrieval significantly outperforms sparse retrieval that uses object expansion. Moreover, dense retrieval matches the performance of sparse retrieval that leverages human-generated captions.

CCS CONCEPTS

• **Information systems** → **Question answering**; **Multimedia and multimodal retrieval**.

KEYWORDS

Dense Retrieval; Multi-Modal; Visual Question Answering

ACM Reference Format:
Chen Qu, Hamed Zamani, Liu Yang, W. Bruce Croft, Erik Learned-Miller. 2021. Passage Retrieval for Outside-Knowledge Visual Question Answering. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '21), July 11–15, 2021, Virtual Event, Canada.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3404835.3462987

1 INTRODUCTION

Recent work on Question Answering (QA) [30, 33, 44] mostly focuses on uni-modal information needs, i.e., text- or voice-based questions (voice input can be considered as text after automatic transcription). However, many information needs, such as the one in Fig. 1, would be inconvenient or hard to explain without a picture. This motivates the study of methods that can handle multi-modal information needs containing both text questions and images [7, 22].

SIGIR '21, July 11–15, 2021, Virtual Event, Canada

ACM ISBN 978-1-4503-8037-9/21/07…$15.00
https://doi.org/10.1145/3404835.3462987

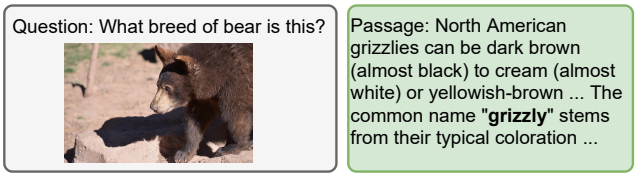


Figure 1: An example of passage retrieval for OK-VQA. Boldface denotes a potential answer. Image ©gsloan, https://www.flickr.com/photos/gsloan/8137199999/

Specifically, we focus on a Visual QA (VQA) task referred to as Outside-Knowledge VQA (OK-VQA). Classic VQA benchmarks [1, 12, 25, 46, 47] and models [3, 10, 17, 23, 26, 41] mainly focus on questions about counting, visual attributes, or other visual detection tasks, whose answers can be found in the given image. In contrast, images in our task help to define the information need, instead of simply being the knowledge source by which the question is answered. OK-VQA resembles open-domain QA [37] in the sense that both tasks require access to an outside and open knowledge resource, e.g., a large collection of passages, to answer the questions. Open-domain QA systems typically follow a retrieve-and-read paradigm [5, 16, 20, 30–32, 43], where the system first retrieves a number of documents (passages) from a collection and then extracts answers from them. This paradigm is less studied for multi-modal information needs, which is the focus of this paper. In this work, we focus on the retrieval phase for OK-VQA as illustrated in Fig. 1.

Unlike previous knowledge-based VQA work that retrieves knowledge from a knowledge base [11, 21, 28, 29, 38–40, 45, 48] or using a Wikipedia Search API [27], we systematically study passage retrieval for OK-VQA with *generic* information retrieval approaches so that our methods can be applied to a wider range of *unstructured* knowledge resources. In particular, we seek answers to the following research questions: (**RQ1**) How helpful are the visual signals in OK-VQA? (**RQ2**) What is the most effective way to incorporate visual signals into sparse retrieval models that are based on term matching? (**RQ3**) How well does dense retrieval [13, 16, 20, 24, 42] work with multi-modal information needs?

To answer these important research questions, we study passage retrieval for OK-VQA queries with a large Wikipedia passage collection. We first conduct sparse retrieval with BM25. We investigate the performance of expanding the original question with different human-annotated object names and image captions. We further study the impact of using different rank fusion methods for different expansion types. We verify that visual clues play an important role in our task. In particular, captions tend to be more informative than object names in sparse retrieval. We further reveal that it is desirable to exploit the most salient matching signal (CombMAX [9, 19])

when using object expansion while it is better to consider the matching signals for all captions with CombSUM [9, 19] or Reciprocal Rank Fusion [14] when we expand with human-generated captions.

We then adopt a dual-encoder architecture to construct a learnable dense retriever following previous work [16, 20, 24, 30, 42]. We employ LXMERT [35], a pre-trained Transformer model [36], as our multi-modal query encoder to encode both the text question and the image as an information need. We observe that our dense retriever achieves a statistically significant performance improvement over sparse retrieval that leverages object expansion, demonstrating the effectiveness of dense retrieval with a multi-modal query encoder. Furthermore, our dense retriever manages to match the performance of sparse retrieval with caption expansion, even though the latter leverages human-generated captions that are often highly informative. Our research is one of fundamental steps for future studies on retrieval-based OK-VQA. Our code is released for research purposes.[1]

2 PASSAGE RETRIEVAL FOR OK-VQA

2.1 Task Definition

We are given an information need (query) denoted as $Q_i = \langle q_i, v_i \rangle$. It consists of a text question q_i and an image v_i. The task is to retrieve k passages that can be used to fulfill Q_i, from a large passage collection. Following the work on open-domain QA [16, 20, 30], a passage is deemed as relevant if it contains the ground truth answer.

2.2 Sparse Retrieval

The backbone of our sparse retrieval approach is BM25, which works with text queries. Therefore, we expand q_i with different textual descriptions of visual clues to construct the BM25 queries. Visual signals in an image are typically expressed in two forms. The first form is a set of object names $\{o_i^1, o_i^2, \cdots\}$ produced by an object detector. Each object is a Region of Interest (RoI) that reveals a meaningful component of the image. The second form is a set of captions $\{c_i^1, c_i^2, \cdots\}$ produced by an image descriptor to describe the image. We adopt human-annotated object names and captions for sparse retrieval. Although the human annotations do not necessarily give the performance upper bound, they would be strong baselines for dense retrieval and make sure that our analysis will not be affected by the quality of automatic annotations produced by object detectors and image descriptors. We study different expansion of visual signals as follows:

- **BM25-Orig**: taking the original q_i only, i.e., $Q_i^{\text{orig}} = \{q_i\}$.
- **BM25-Obj** (object expansion): appending each one of the object names to q_i, i.e., $Q_i^{\text{obj}} = \{q_i + o_i^1, q_i + o_i^2, \cdots\}$.
- **BM25-Cap** (caption expansion): appending each one of the captions to q_i, i.e., $Q_i^{\text{cap}} = \{q_i + c_i^1, q_i + c_i^2, \cdots\}$.
- **BM25-All**: taking the union of the above queries , i.e., $Q_i^{\text{all}} = Q_i^{\text{orig}} \cup Q_i^{\text{obj}} \cup Q_i^{\text{cap}}$.

Since $Q_i^{\text{obj/cap/all}}$ contains multiple BM25 queries for the same information need Q_i, we need rank fusion methods to consolidate the ranked lists R generated by queries within each query set. This resembles an ensemble process to combine results obtained with different visual signals. We consider **CombMAX** [9, 19] (taking the maximum score of a passage in different ranked lists), **CombSUM** [9, 19] (taking the sum of scores of a passage in different ranked lists), and **RRF** (Reciprocal Rank Fusion) [6] (the fusion score for a passage p is defined as $\sum_{r \in R} \frac{1}{const + r(p)}$, where $r(\cdot)$ is the rank of p). CombMAX could help the model be more robust to distracting visual signals while the other two fusion approaches make sure that the impacts of lower-ranked passages do not vanish.

[1] https://github.com/prdwb/okvqa-release

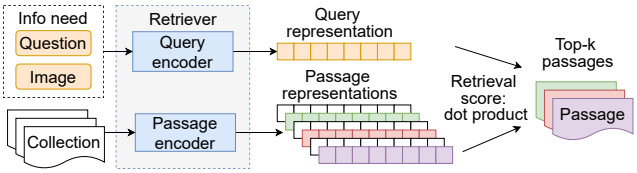


Figure 2: Dense retrieval with neural dual encoders.

2.3 Dense Retrieval

Following previous work [16, 20, 30, 42], we adopt a dual-encoder architecture to construct a learnable retriever. The retrieval process is "dense" in the sense that the queries and passages are encoded to low-dimensional dense vectors, as opposed to the high-dimensional sparse vectors used in sparse retrieval. As shown in Fig. 2, the retriever consists of a query encoder and a passage encoder.

2.3.1 ***Query Encoder.*** We adopt the LXMERT model [35] as the query encoder since it can encode both the question and image components of Q_i. LXMERT is a pre-trained Transformer model [36] designed to learn vision and language connections. It consists of three encoders, an object relationship encoder, a language encoder, and a cross-modality encoder. The first two single-modality encoders function in a similar way to BERT [8] except that the object relationship encoder works with a set of object detections produced by a Faster R-CNN [34] model pre-trained on Visual Genome [2, 18]. Each detection representation can be considered as an "image token embedding" that consists of its RoI features (fixed) and position features (trainable). The cross-modality encoder conducts bi-directional cross attention between vision and language representations. We refer our readers to the LXMERT paper [35] for further details. We project the cross-modality output of LXMERT to an n-dimensional query representation. The dense retriever with a LXMERT query encoder is referred to as **Dense-LXMERT**. To adopt a deeper analytical view, we also consider BERT as the query encoder, which only works with the question component of the query, resulting in the **Dense-BERT** model.

2.3.2 ***Passage Encoder.*** We use BERT as the passage encoder and project the `[CLS]` representation to an n-dimensional passage representation. The retrieval score is defined as the dot product of the query and passage representations. After training, we encode all passages in the collection during an offline process. At inference time, we use Faiss [15] for maximum inner product search.

2.3.3 ***Training.*** We train the dual-encoder retriever with a set of training instances. Each instance is denoted as $\langle Q_i, p_i^+, p_i^- \rangle$, where p_i^+ is a positive passage that contains the answer while p_i^- is a negative passage that does not contain the answer. In our case,

we select p_i^- from the top passages retrieved by a sparse retrieval method. Thus, p_i^- can be referred to as a *retrieved negative*. We present more details on constructing the training data in Sec. 3.1.2. In addition to the retrieved negatives, one might want to take advantage of the other passages in the batch as in-batch negatives. Although in-batch negatives resemble randomly sampled negatives that can be less effective [16], it is extremely efficient since passage representations can be reused within the batch. Karpukhin et al. [16] studied combining in-batch negatives with retrieved negatives for uni-modal queries. We further dig into this topic for multi-modal queries. We consider the following negative sampling strategies:

- **R-Neg**: using the **r**etrieved **neg**ative passage only.
- **R-Neg+IB-Neg**: using the **r**etrieved **neg**ative, along with all other **i**n-**b**atch **neg**ative passages of other instances.
- **R-Neg+IB-Pos**: using the **r**etrieved **neg**ative, along with all other **i**n-**b**atch **pos**itive passages of other instances.
- **R-Neg+IB-All**: using the **r**etrieved **neg**ative, along with **all** other **i**n-**b**atch passages, except for p_i^+. The same query can be paired with different positive and negative passages to augment the training data as suggested in Sec. 3.1.2. Therefore, the queries within a batch can coincide even with random batching. In this case, the misuse of a positive passage as negative may hinder the learning process. We empirically examine whether this concern holds by comparing R-Neg+IB-All/Pos to R-Neg+IB-Neg.

Following previous work [16, 30], we use cross entropy loss to maximize the probability of the positive passage given the negatives identified above. We then average the losses for queries in the batch.

3 EXPERIMENTS

3.1 Experimental Setup

3.1.1 ***Dataset.*** Our retrieval dataset is based on the OK-VQA dataset [27], where all questions require outside knowledge.[2] The images in the OK-VQA dataset are from the COCO dataset [12]. We take the original training queries as our training queries and split the original validation queries into our validation and testing queries. In terms of the collection, we take the Wikipedia passage collection with 11 million passages created by previous work [30].[3] Each passage contains at most 384 "wordpieces" [8] with intact sentence boundaries. Data statistics are presented in Tab. 1.

3.1.2 ***Data Construction for Dense Retrieval.*** We create the training instances described in Sec. 2.3.3 using retrieved passages of sparse retrieval (see configuration details in Sec. 3.1.4). A passage is identified to be positive if it contains an exact match (case-insensitive) of a ground truth answer. The other retrieved passages are considered as negatives. We take the top 5 positive passages, each repeated 5 times (for augmentation), to construct training instances with random retrieved negatives. In addition, we put together a small validation collection by taking the top 20 passages for each question. Data statistics are presented in Tab. 1.

3.1.3 ***Evaluation Metrics.*** We focus on passage retrieval for multi-modal information needs as the first step in the OK-VQA pipeline. The output of the retrieval process will be used by a reader model to extract the answer. Therefore, following previous work [5, 16], we use precision-oriented metrics to evaluate the performance of our models. Precisely, we use Mean Reciprocal Rank and Precision with the ranking cut-off of 5 (MRR@5 and P@5) as our metrics.

Table 1: Data statistics.

Split	#. questions	#. BM25 queries	#. training instances	#. passages in collection
Train	9,009	81,100	211,200	N/A
Val	2,523	22,352	N/A	34,059
Test	2,523	22,573	N/A	11,000,000

3.1.4 ***Implementation Details.*** For sparse retrieval, we use BM25 in Anserini (v0.5.1).[4] We tune $k_1 \in [0.5, 1.5]$ and $b \in [0.2, 0.8]$ with steps of 0.2 based on validation MRR. The best setting is $k_1 = 1.1, b = 0.4$. The constant in RRF is set to 60 [4, 6]. Human-annotated object names and captions are from the COCO dataset [12]. For dense retrieval, we use the HuggingFace transformers library[5] for the implementations of LXMERT and BERT. We set the maximum sequence length of the query encoder to 20 [35], that of the passage encoder to 384, the projection size (n) for the query/passage representations to 768, the learning rate to 1e-5, the batch size to 4 per GPU, and the number of fine-tuning epochs to 2. We adopt R-Neg+IB-All as the negative sampling strategy. We save checkpoints every 5,000 steps and evaluate on the validation set to select the best model for the test set. The training time is 10 hours for Dense-BERT and 12 hours for Dense-LXMERT. All models are trained with 4 GPUs with mixed-precision training. Warm-up takes 10% of the total steps. The training instances are constructed with the top 100 retrieved passages for each question using BM25-Cap (CombSUM) with the default BM25 configuration in Anserini ($k_1 = 0.9, b = 0.4$).

3.2 Main Results

3.2.1 ***Sparse Retrieval.*** We present the results for sparse retrieval in Tab. 2 to answer **RQ1** and **RQ2** raised in Sec. 1. First, we observe that approaches that consider visual signals outperform BM25-Orig by a large margin, verifying that visual clues are helpful in our task. We then compare BM25 with different forms of visual clues. Methods with captions (BM25-Cap/All) outperform object expansion, indicating that captions are more informative than object names. This makes sense since captions typically cover important objects descriptively. BM25-All does not benefit from incorporating both objects and captions. On the contrary, objects can be distracting and hurt the performance gain from captions. Finally, we compare different rank fusion methods. When objects are being considered (BM25-Obj/All), CombMAX yields the best performance since it is robust to potentially misleading objects by only considering objects with the best matching score. On the other hand, CombSUM and RRF work well with caption expansion. Their ability to consider the impact of all captions is desirable since captions are closely connected to the image and can be diverse and complementary. The best performing approach is BM25-Cap with CombSUM.

3.2.2 ***Dense Retrieval.*** We present the dense retrieval results, along with the best sparse retrieval results in Tab. 3 to answer **RQ3**. We first compare retrieval without visual clues: we observe that Dense-BERT outperforms BM25-Orig by a large margin, verifying

[2] https://okvqa.allenai.org/index.html
[3] https://ciir.cs.umass.edu/downloads/ORConvQA/all_blocks.txt.gz
[4] https://github.com/castorini/anserini
[5] https://github.com/huggingface/transformers

Table 2: Sparse retrieval results. Boldface denotes the best performance within each group and underscores denote the best overall results. ▲i denotes that the gain with respect to the best method in group i has statistically significance with $p < 0.05$ tested by the Student's paired t-test.

Methods		Val		Test	
Expansion	Fusion	MRR@5	P@5	MRR@5	P@5
1. BM25-Orig	N/A	0.2565	0.1772	0.2637	0.1755
2. BM25-Obj	CombMAX	**0.3772**$^{\blacktriangle 1}$	**0.2667**$^{\blacktriangle 1}$	**0.3686**$^{\blacktriangle 1}$	**0.2541**$^{\blacktriangle 1}$
	CombSUM	0.3493	0.2395	0.3406	0.2322
	RRF	0.3389	0.2291	0.3292	0.2213
3. BM25-Cap	CombMAX	0.4547	0.3294	0.4534	0.3230
	CombSUM	**0.4727**$^{\blacktriangle 1,2,4}$	**0.3483**$^{\blacktriangle 1,2,4}$	**0.4622**$^{\blacktriangle 1,2}$	**0.3367**$^{\blacktriangle 1,2,4}$
	RRF	0.4689	0.3440	0.4585	0.3346
4. BM25-All	CombMAX	**0.4550**$^{\blacktriangle 1,2}$	**0.3293**$^{\blacktriangle 1,2}$	**0.4533**$^{\blacktriangle 1,2}$	**0.3233**$^{\blacktriangle 1,2}$
	CombSUM	0.4490	0.3241	0.4396	0.3126
	RRF	0.4322	0.3069	0.4260	0.2956

Table 3: Dense retrieval results. Refer to Tab. 2 for notations. $\triangle i$ denotes the statistical significance is obtained with $0.05 < p < 0.1$. Note that *BM25-Obj/Cap/All has access to the ground truth object names and captions.*

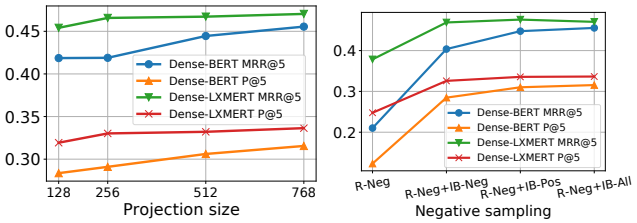
	Methods	Val		Test	
		MRR@5	P@5	MRR@5	P@5
Sparse	1. BM25-Orig	0.2565	0.1772	0.2637	0.1755
	2. BM25-Obj (CombMAX)	0.3772$^{\blacktriangle 1}$	0.2667$^{\blacktriangle 1}$	0.3686$^{\blacktriangle 1}$	0.2541$^{\blacktriangle 1}$
	3. BM25-Cap (CombSUM)	**0.4727**$^{\blacktriangle 1,2,4\ \triangle 5}$	**0.3483**$^{\blacktriangle 1,2,4,5}$	**0.4622**$^{\blacktriangle 1,2,5}$	**0.3367**$^{\blacktriangle 1,2,4,5}$
	4. BM25-All (CombMAX)	0.4550$^{\blacktriangle 1,2}$	0.3293$^{\blacktriangle 1,2\ \triangle 5}$	0.4533$^{\blacktriangle 1,2\ \triangle 5}$	0.3233$^{\blacktriangle 1,2,5}$
Dense	5. Dense-BERT	0.4555$^{\blacktriangle 1,2}$	0.3155$^{\blacktriangle 1,2}$	0.4325$^{\blacktriangle 1,2}$	0.3058$^{\blacktriangle 1,2}$
	6. Dense-LXMERT	**0.4704**$^{\blacktriangle 1,2\ \triangle 5}$	**0.3364**$^{\blacktriangle 1,2,5}$	**0.4526**$^{\blacktriangle 1,2,5}$	**0.3329**$^{\blacktriangle 1,2,5}$

the capability of dense retrieval. We further explain that this capability is contingent upon the negative sampling strategy used during training in Sec. 3.3.2. Moreover, Dense-BERT even surpasses BM25-Obj that considers visual signals. This could be due to the tendency of Dense-BERT to retrieve passages containing frequent answers. We speculate this kind of overfitting is caused by the lack of visual signals. Further analysis can be found in Sec. 3.3.

We further observe that Dense-LXMERT significantly outperforms BM25-Obj. Dense-LXMERT leverages both the RoI features and position features in object detection to learn object relations, which can be more effective than using ground truth object names in sparse retrieval. We then compare Dense-LXMERT with BM25-Cap/All. These sparse retrieval methods consider human-annotated image captions that are highly informative and descriptive. On the contrary, Dense-LXMERT has to learn the importance of the objects and the relation among them with object-level features. In this unfavorable situation, Dense-LXMERT still manages to match the performance of BM25-Cap/All. Although BM25-Cap is slightly better, the margins are statistically *insignificant*. Finally, we observe that Dense-LXMERT significantly outperforms Dense-BERT, further validating the use of a multi-modal query encoder.

3.3 Additional Results

We provide additional analysis to study the impact of the projection size and negative sampling strategies with validation performance.



(a) Impact of projection size n. **(b) Impact of negative sampling.**

Figure 3: Additional results.

3.3.1 ***Impact of projection size.*** We present the impact of the dimensionality n of query/passage representations in Fig. 3a. We observe that a larger projection size always leads to better performance, although the performance gain seems to be insignificant for LXMERT after $n = 256$. We set $n = 768$ as reported in Sec.3.1.4 since it gives the best performance for both Dense-LXMERT and Dense-BERT. When working with a much larger collection than ours (ours has 11 million passages), one might want to use $n = 256$ since it offers similar performance with less memory consumption.

3.3.2 ***Impact of negative sampling.*** The desirable performance of dense retrieval is contingent upon the negative sampling strategy. We present the impact of different sampling methods described in Sec. 2.3.3 in Fig. 3b. We observe that combining retrieved negatives with in-batch negatives dramatically improves the model performance, verifying the observations in Karpukhin et al. [16] for multi-modal queries. Also, different choices of in-batch negatives (R-Neg+IB-Neg/Pos/All) give a similar performance for LXMERT, indicating that coinciding questions in the same batch should not be a concern for our batch size and data size reported in Sec. 3.1.

Both analyses show that Dense-BERT is more demanding on larger model capacity (larger projection size) and more negative samples. We speculate that BERT is overfitting the patterns in the training data since it lacks important visual clues for matching. In comparison, Dense-LXMERT is less sensitive to reasonably-chosen projection sizes and negative sampling strategies because it can learn matching signals from both language and vision clues.

4 CONCLUSIONS AND FUTURE WORK

We study passage retrieval for OK-VQA with sparse and dense retrieval and verify visual clues play an important role. We discover that captions are more informative than object names in sparse retrieval and CombMAX works well with object expansion while CombSUM and RRF are better for caption expansion. We further show a dense retriever with a multi-modal query encoder can significantly outperform sparse retrieval with object expansion and even matches the performance of that with human-generated captions. In the future, we will consider using automatic captions for sparse retrieval and study answer extraction to complete the QA pipeline.

ACKNOWLEDGMENTS

This work was supported in part by the Center for Intelligent Information Retrieval. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect those of the sponsors.

REFERENCES

[1] A. Agrawal, J. Lu, S. Antol, M. Mitchell, C. L. Zitnick, D. Parikh, and D. Batra. VQA: Visual Question Answering. *International Journal of Computer Vision*, 123: 4–31, 2015.

[2] P. Anderson, X. He, C. Buehler, D. Teney, M. Johnson, S. Gould, and L. Zhang. Bottom-up and top-down attention for image captioning and visual question answering. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6077–6086, 2018.

[3] H. Ben-younes, R. Cadène, M. Cord, and N. Thome. MUTAN: Multimodal Tucker Fusion for Visual Question Answering. In *ICCV*, 2017.

[4] R. Benham and J. S. Culpepper. Risk-Reward Trade-offs in Rank Fusion. *Proceedings of the 22nd Australasian Document Computing Symposium*, 2017.

[5] D. Chen, A. Fisch, J. Weston, and A. Bordes. Reading Wikipedia to Answer Open-Domain Questions. In *ACL*, 2017.

[6] G. V. Cormack, C. L. A. Clarke, and S. Büttcher. Reciprocal rank fusion outperforms condorcet and individual rank learning methods. In *SIGIR*, 2009.

[7] Y. Deldjoo, J. R. Trippas, and H. Zamani. Towards Multi-Modal Conversational Information Seeking. In *SIGIR*, 2021.

[8] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *NAACL-HLT*, 2019.

[9] E. A. Fox and J. A. Shaw. Combination of Multiple Searches. In *TREC*, 1993.

[10] A. Fukui, D. H. Park, D. Yang, A. Rohrbach, T. Darrell, and M. Rohrbach. Multimodal Compact Bilinear Pooling for Visual Question Answering and Visual Grounding. In *EMNLP*, 2016.

[11] F. Gardères, M. Ziaeefard, B. Abeloos, and F. Lécué. ConceptBert: Concept-Aware Representation for Visual Question Answering. In *EMNLP*, 2020.

[12] Y. Goyal, T. Khot, D. Summers-Stay, D. Batra, and D. Parikh. Making the V in VQA Matter: Elevating the Role of Image Understanding in Visual Question Answering. In *CVPR*, 2017.

[13] K. Guu, K. Lee, Z. Tung, P. Pasupat, and M.-W. Chang. REALM: Retrieval-Augmented Language Model Pre-Training. *ArXiv*, abs/2002.08909, 2020.

[14] M. Hu, Y. Peng, Z. Huang, X. Qiu, F. Wei, and M. Zhou. Reinforced Mnemonic Reader for Machine Reading Comprehension. In *IJCAI*, 2018.

[15] J. Johnson, M. Douze, and H. Jégou. Billion-scale similarity search with GPUs. *ArXiv*, 2017.

[16] V. Karpukhin, B. Oğuz, S. Min, P. Lewis, L. Y. Wu, S. Edunov, D. Chen, and W. tau Yih. Dense Passage Retrieval for Open-Domain Question Answering. In *EMNLP*, 2020.

[17] J. Kim, J. Jun, and B. Zhang. Bilinear Attention Networks. In *NeurIPS*, 2018.

[18] R. Krishna, Y. Zhu, O. Groth, J. Johnson, K. Hata, J. Kravitz, S. Chen, Y. Kalantidis, L. Li, D. A. Shamma, M. S. Bernstein, and L. Fei-Fei. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision*, 123:32–73, 2016.

[19] J. H. Lee. Analyses of Multiple Evidence Combination. In *SIGIR*, 1997.

[20] K. Lee, M.-W. Chang, and K. Toutanova. Latent Retrieval for Weakly Supervised Open Domain Question Answering. In *ACL*, 2019.

[21] G. Li, H. Su, and W. Zhu. Incorporating external knowledge to answer open-domain visual questions with dynamic memory networks. *ArXiv*, abs/1712.00733, 2017.

[22] Y.-C. Lien, H. Zamani, and W. B. Croft. Recipe Retrieval with Visual Query of Ingredients. In *SIGIR*, 2020.

[23] J. Lu, J. Yang, D. Batra, and D. Parikh. Hierarchical Question-Image Co-Attention for Visual Question Answering. In *NIPS*, 2016.

[24] Y. Luan, J. Eisenstein, K. Toutanova, and M. Collins. Sparse, Dense, and Attentional Representations for Text Retrieval. *ArXiv*, abs/2005.00181, 2020.

[25] M. Malinowski and M. Fritz. A Multi-World Approach to Question Answering about Real-World Scenes based on Uncertain Input. In *NIPS*, 2014.

[26] M. Malinowski, M. Rohrbach, and M. Fritz. Ask Your Neurons: A Neural-Based Approach to Answering Questions about Images. In *ICCV*, pages 1–9, 2015.

[27] K. Marino, M. Rastegari, A. Farhadi, and R. Mottaghi. OK-VQA: A Visual Question Answering Benchmark Requiring External Knowledge. In *CVPR*, 2019.

[28] M. Narasimhan and A. G. Schwing. Straight to the Facts: Learning Knowledge Base Retrieval for Factual Visual Question Answering. In *ECCV*, 2018.

[29] M. Narasimhan, S. Lazebnik, and A. G. Schwing. Out of the Box: Reasoning with Graph Convolution Nets for Factual Visual Question Answering. In *NeurIPS*, 2018.

[30] C. Qu, L. Yang, C. Chen, M. Qiu, W. B. Croft, and M. Iyyer. Open-Retrieval Conversational Question Answering. In *SIGIR*, 2020.

[31] C. Qu, L. Yang, C. Chen, W. Croft, K. Krishna, and M. Iyyer. Weakly-Supervised Open-Retrieval Conversational Question Answering. In *ECIR*, 2021.

[32] Y. Qu, Y. Ding, J. Liu, K. Liu, R. Ren, X. Zhao, D. Dong, H. Wu, and H. Wang. RocketQA: An Optimized Training Approach to Dense Passage Retrieval for Open-Domain Question Answering. *ArXiv*, abs/2010.08191, 2020.

[33] P. Rajpurkar, J. Zhang, K. Lopyrev, and P. Liang. SQuAD: 100, 000+ Questions for Machine Comprehension of Text. In *EMNLP*, 2016.

[34] S. Ren, K. He, R. B. Girshick, and J. Sun. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39:1137–1149, 2015.

[35] H. H. Tan and M. Bansal. LXMERT: Learning Cross-Modality Encoder Representations from Transformers. In *EMNLP/IJCNLP*, 2019.

[36] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. Attention Is All You Need. In *NIPS*, 2017.

[37] E. M. Voorhees and D. M. Tice. The TREC-8 Question Answering Track Evaluation. In *TREC*, 1999.

[38] P. Wang, Q. Wu, C. Shen, A. Dick, and A. V. D. Hengel. Explicit Knowledge-based Reasoning for Visual Question Answering. In *IJCAI*, 2017.

[39] P. Wang, Q. Wu, C. Shen, A. Dick, and A. van den Hengel. FVQA: Fact-Based Visual Question Answering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40:2413–2427, 2018.

[40] Q. Wu, P. Wang, C. Shen, A. Dick, and A. V. D. Hengel. Ask Me Anything: Free-Form Visual Question Answering Based on Knowledge from External Sources. In *CVPR*, 2016.

[41] C. Xiong, S. Merity, and R. Socher. Dynamic Memory Networks for Visual and Textual Question Answering. In *ICML*, 2016.

[42] L. Xiong, C. Xiong, Y. Li, K.-F. Tang, J. Liu, P. Bennett, J. Ahmed, and A. Overwijk. Approximate Nearest Neighbor Negative Contrastive Learning for Dense Text Retrieval. *ArXiv*, abs/2007.00808, 2020.

[43] W. Xiong, X. Li, S. Iyer, J. Du, P. Lewis, W. Y. Wang, Y. Mehdad, W. tau Yih, S. Riedel, D. Kiela, and B. Ouguz. Answering Complex Open-Domain Questions with Multi-Hop Dense Retrieval. *ArXiv*, abs/2009.12756, 2020.

[44] L. Yang, M. Qiu, C. Qu, C. Chen, J. Guo, Y. Zhang, W. B. Croft, and H. Chen. IART: Intent-Aware Response Ranking with Transformers in Information-Seeking Conversation Systems. In *WWW*, 2020.

[45] J. Yu, Z. Zhu, Y. Wang, W. Zhang, Y. Hu, and J. Tan. Cross-modal Knowledge Reasoning for Knowledge-based Visual Question Answering. *Pattern Recognition*, 108:107563, 2020.

[46] L. Yu, E. Park, A. Berg, and T. Berg. Visual Madlibs: Fill in the Blank Description Generation and Question Answering. In *ICCV*, 2015.

[47] Y. Zhu, O. Groth, M. S. Bernstein, and L. Fei-Fei. Visual7W: Grounded Question Answering in Images. In *CVPR*, 2016.

[48] Z. Zhu, J. Yu, Y. Wang, Y. Sun, Y. Hu, and Q. Wu. Mucko: Multi-Layer Cross-Modal Knowledge Reasoning for Fact-based Visual Question Answering. In *IJCAI*, 2020.