

WDRASS: A Web-scale Dataset for Document Retrieval and Answer Sentence Selection

Zeyu Zhang
University of Arizona
Tucson, Arizona, USA
zeyuzhang@email.arizona.edu

Thuy Vu
Amazon Alexa AI
Manhattan Beach, California, USA
thuyvu@amazon.com

Sunil Gandhi
Amazon Alexa AI
Seattle, Washington, USA
gansuni@amazon.com

Ankit Chadha
Amazon Alexa AI
Sunnyvale, California, USA
ankitrc@amazon.com

Alessandro Moschitti
Amazon Alexa AI
Manhattan Beach, California, USA
amosch@amazon.com

ABSTRACT

Open-Domain Question Answering (ODQA) systems generate answers from relevant text returned by search engines, e.g., lexical features-based such as BM25, or embeddings-based such as dense passage retrieval (DPR). Few datasets are available for this task: they mainly focus on QA systems based on machine reading (MR) approach, and show problematic evaluation, mostly based on un-contextualized short answer matching. In this paper, we present WDRASS, a dataset for ODQA based on answer sentence selection (AS2) models, which consider sentences as candidate answers for QA systems. WDRASS consists of ~64k questions and 800k+ labeled passages and sentences extracted from 30M documents. We evaluate the dataset by training models on it and comparing with the same models trained on Google NQ. Our experiments show that WDRASS significantly improves the performance of retrieval and reranking models, thus boosting the accuracy of downstream QA tasks. We believe our dataset can produce significant impact in advancing IR research.

CCS CONCEPTS

• Information systems → Question answering.

KEYWORDS

machine learning, information retrieval, question answering

ACM Reference Format:

Zeyu Zhang, Thuy Vu, Sunil Gandhi, Ankit Chadha, and Alessandro Moschitti. 2022. WDRASS: A Web-scale Dataset for Document Retrieval and Answer Sentence Selection. In *Proceedings of the 31st ACM Int'l Conference on Information and Knowledge Management (CIKM '22)*, Oct. 17–21, 2022, Atlanta, GA, USA. ACM, New York, NY, USA, 5 pages. <https://doi.org/10.1145/3511808.3557678>

1 INTRODUCTION

ODQA systems compute answer for a question through a pipeline consisting of (i) a text retriever, which returns content relevant to a question, i.e., documents or passages, from a text collection, and (ii)

factoid question

q_1 : What is the population of California?
 a_1 : The resident population of California has been steadily increasing over the past few decades and has increased to **39.56 million** people in 2018.
 s_1 : **39.56 million**

non-factoid question

q_2 : What makes tides?
 a_2 : Tides are formed by a combination of gravitational attraction from the moon and the sun, as well as the centrifugal force generated by the rotation of the Earth.

Table 1: For factoid question q_1 , it is possible to locate a short span s_1 from the sentence; yet, there can be infinite valid variants, e.g., 39.6 million or 40 million. For non-factoid question q_2 , it requires the whole sentence to answer the question.

a QA module, e.g., machine reading (MR) or answer sentence selection (AS2), which extracts answer from the relevant content. ODQA is scoped to be applicable at Web scale QA, i.e., answering *any* questions using content available in the Web [24, 25]. Most previous work, however, is limited to (i) content from only Wikipedia and (ii) often a small and not representative question set. In addition, recent work mainly focuses on MR, which shows limitations in the type of questions and answers it can deal with. Indeed, although MR-based systems can potentially select longer answers spanning entire paragraphs, they are in practice optimized for the retrieval and selection of named entities (or sometimes short phrases). Thus, they mainly solve factoid QA tasks.

In contrast, AS2-based systems are optimized on the entire sentence semantics. This makes them suitable for processing both factoid and non-factoid questions (see Table 1). The former are typically answerable by a short span of text, i.e., nouns, phrases, or entities, in a sentence or a passage, which provide the context. The latter, instead, require an elaborated response, typically a full sentence or longer text, rather than a short span. This is typically a large class, e.g., in Google's Natural Question (NQ) dataset [14], two third of the questions are non-factoid: they have no answer span. However, as MR has been the main target of DPR, the average answer length of the DPR test set [12] is only 2.3 with a max of 4 words. This has produced research on neural retrieval suitable for MR-based systems [12, 16], but sub-optimal for general QA. Also,



This work is licensed under a Creative Commons Attribution International 4.0 License.

CIKM '22, October 17–21, 2022, Atlanta, GA, USA
© 2022 Copyright held by the owner/author(s).
ACM ISBN 978-1-4503-9236-5/22/10.
<https://doi.org/10.1145/3511808.3557678>

the created DPR datasets fit the MR task [11, 16, 20]: the question and passage relation is mainly defined by an answer span.

Finally, this research produces suboptimal retrieval models as: (i) a large amount of false negatives are produced since there can be infinite variants of a correct answer span, thus they cannot all be captured in the golden set, and (ii) a large number of false positives is also generated since the presence of a correct answer span in a passage does not guarantee the passage is relevant [18].

This paper contributes to the research community proposing an ODQA benchmark that addresses all the aforementioned shortcomings. We propose WDRASS, a Web-scale Document Retrieval and Answer Sentence Selection dataset, which enables the training of AS2 models at Web-scale. Using the AS2 component, we define a passage to be relevant to a question if it contains an entire correct answer sentence for the question. This definition is highly conservative in comparison to previous datasets, as it requires direct answerability relation between a question and a passage. Formally, let p be a passage consisting of n sentences $p = \{s_1, s_1, \dots, s_n\}$. We define a question-passage pair (q, p) relevant if there exists $s_i \in p$ such that (q, s_i) is a correct question-answer pair.

Our contribution is two-fold: First, we construct WDRASS, a large dataset for ODQA research, which includes passage retrieval and answer reranking. We show that the dataset can add significant supervision both to factoid and non-factoid, improving the AS2 accuracy by 8.21% and 6.89% on each question type, respectively, according to an end-end human evaluation. Second, we design an optimized setting that efficiently combines the strengths of both two-tower model [12] and pair-wise model [21]. We show that our design better captures question/passage relevance.

We measure the performance of our proposed system end-to-end, with a large index having up to 130M passages from CommonCrawl. To the best of our knowledge, this is the largest index size ever reported for neural-based retrieval studies.

In the reminder, Sec. 2 reports the related work. Sec. 3 shows details for the construction of the dataset. Sec. 4 presents our experimental studies. Finally, we conclude our work in Sec. 5.

2 RELATED WORK AND BACKGROUND

Embedding-based retrieval (dense retrieval) systems measure text similarity between query and text using their vector representations, i.e., embeddings. The latter are learned following two main approaches: (i) self-supervised pre-training or (ii) supervised fine-tuning on labeled question-passage pairs. ORQA [16] is an example of the former, which firstly achieved a better performance than standard lexical-based methods, e.g., BM25 [5]. Since then, several variants have been proposed. REALM [9] uses salient masked language model (MLM) for pre-training the retriever. DPR achieves significant improvement on MR task using encoders for question and passage, trained using labeled question-passage pairs. ROM[7] extended this work to multi-task setting by training the neural retriever using scheduled learning. RAG [17] and Fusion-in-decoder [10] used DPR with a generative model embedded in a MR for ODQA.

This research work has also generated several datasets: WebAP [13], WikiPassageQA [6], TriviaQA [11]. However, they target an MR setting, where the majority of questions are of factoid type. This produces suboptimal retrieval performance when retrieving information for non-factoid questions.

In contrast, we focus on improving dense retrieval for AS2, which can be used to build ODQA systems, as well as other applications using passage/sentence retrieval. AS2 can operate on any question, both factoid and non-factoid, and can be considered the supervised version of passage reranking, even though the size of the passage tend to be of one single sentence. Datasets from previous work for AS2 are rather small, e.g., [25, 26] and cannot be used for retrieval.

2.1 Embedding-based Text Retrieval

The retrieval setting aims at finding most similar text for a given query by their embedding-based similarity. Formally, the approach builds two encoders $E_q(\cdot)$ and $E_p(\cdot)$ for question and passage to map question and text, respectively, to a d -dimensional real-valued representation. The similarity between the question and the passage is the dot product of their vectors: $\text{sim}(q, p) = E_q(q)^\top E_p(p)$.

The encoder is trained to make the dot-product reflect the expected ranking. For this purpose, the similarity between a question and relevant passages is maximized, while it is minimized for irrelevant passages¹. Let $\mathcal{D} = \{(q_i, p_i^+, p_{i,1}^-, \dots, p_{i,n}^-)\}_{i=1}^m$ be the training data of m examples. Each example consists of a question q_i , a relevant passage p_i^+ , and a set of n irrelevant passages $p_{i,j}^-$. The loss function is defined as follows:

$$L(q_i, p_i^+, p_{i,1}^-, \dots, p_{i,n}^-) = -\log \frac{e^{\text{sim}(q_i, p_i^+)}}{e^{\text{sim}(q_i, p_i^+)} + \sum_{j=1}^n e^{\text{sim}(q_i, p_{i,j}^-)}}.$$

2.2 Transformer-based Text Reranking

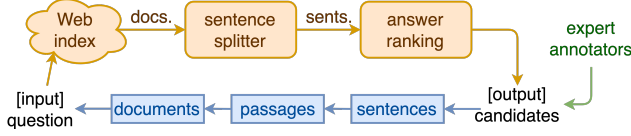
Text ranking with respect a question can be modeled using pointwise [21, 27], pairwise [15, 19, 23], and listwise [1–4] methods. We focus on pointwise as it is simple and achieves state-of-the-art results [8]. Let q be an element of the question set, Q , and $\mathcal{A} = \{c_1, \dots, c_n\}$ be a set of candidates for q , a reranker can be defined as $\mathcal{R} : Q \times \Pi(\mathcal{A}) \rightarrow \Pi(\mathcal{A})$, where $\Pi(\mathcal{A})$ is the set of all permutations of $\mathcal{A}$. Previous work estimates $p(q, c_i)$ with neural models [21], also using attention mechanisms, e.g., Compare-Aggregate [27], inter-weighted alignment networks [22], and pre-trained Transformer models. Garg et al. [8] proposed TANDA, which is the state of the art on the WikiQA and TREC-QA datasets. This is based on a classifier based on transformer model, which is first fine-tuned on NQ and then on the target dataset.

3 RETRIEVAL AND RERANKING DATASET

We describe our approach to build (WDRASS) for ODQA exploiting annotation at sentence level, i.e., AS2 labelled data. In retrieval-based ODQA answers to a question can be extracted from any Web document. Thus, we built an ODQA pipeline that enabled the selection of relevant answer candidates from a set of documents retrieved from a large Web index. The questions and their top candidate answers are validated by expert annotators, judging the quality of questions and the correctness of the candidates. More in detail, the construction of WDRASS follows four steps: creation of *Web Document Collection and Index*, *Question Sampling and Annotation*, *Answer Sentence Annotation* using our *Annotation Criteria*, and finally, *Transferring Labels to Passages and Documents*.

¹Hard-negative are selected as negative examples.

Figure 1: ODQA pipeline for generating question-candidate pairs, which are labelled by expert annotators (upper/orange pipeline). The annotation is propagated back to the embedding passages and documents (lower/blue pipeline).



Web Document Collection and Index. We built a large collection of Web data, i.e., documents and passages, to facilitate data construction and end-to-end system evaluation. This resource allows us to measure the impact of our work in an industry-scale ODQA setting. We selected English Web documents of the 5,000 most popular domains, including Wikipedia, from releases of Common Crawl in 2019 and 2020. The pages are then split into passages following the DPR procedure [12]. We limit the length of a passage to 200 tokens while preserving sentence boundaries. This process produced a collection of $\sim 100\text{M}$ and $\sim 130\text{M}$ of documents and passages, respectively. We built (i) a standard index using Lucene/Elasticsearch and (ii) a neural-based index using DPR [12] from the collection.

Question Sampling and Annotation. We extracted 64,230 questions from the collection above. For this purpose, we used efficient and simple heuristics to find questions and apply automatic classifiers to filter out unintelligible ones. Then, we used annotators to manually refine the final set². They also annotated (i) question type, i.e., factoid and non-factoid, and (ii) quality, i.e., well-formed or grammatically correct.

Answer Sentence Annotation. For each question, we retrieved up to 1,000 documents/passages using both indexes. We split them into sentences and applied a public state-of-the-art AS2 model [8] to rank the sentences. We only annotated the top 15 candidate sentences with the positive labels if they correctly and sufficiently answer the question. As we aim at high-quality data, a correct answer must address the inquired information and also be conversational and direct. In total, we collected more than 800K pairs.

Annotation Criteria. The annotation for this task is highly ambiguous and challenging given the scope of ODQA with content from a large Web collection. We enforced annotation quality in four steps: (i) Annotators are required to pass a qualitative test with a set of golden labels; (ii) Each annotation for both questions and answers is judged by two annotators. (iii) The annotators are encouraged to verify the answers using web searches when needed. (iv) In case of conflicting decisions, a senior annotator serves as a tiebreaker. The accuracy of answer annotation with respect to the senior annotator is 95%. We also verified the annotation quality for a sample of 15K with an independent annotation team, assessing an accuracy of 90%.

Transferring Labels to Passages and Documents. We simply propagate a positive label to all passages and documents that contain a positive answer sentence.

Table 2 reports WDRASS statistics: number of questions, documents, and passages subdivided in training, dev., and test sets, along with the number of positive and negative passages and answers.

²Around 2% of the questions were found not well-formed or grammatically correct yet interpretable.

Split	#Quest.	#Docs	#Passages			#Answers		
			#QP Pairs	#QP+	#QP-	#QA Pairs	#QA+	#QA-
Train	53,419	18,983,495	695,033	265,177	429,856	695,130	265,217	429,913
Dev	5,416	6,015,441	70,981	26,989	43,992	70,988	26,992	43,996
Test	5,395	6,042,851	70,365	26,574	43,791	70,372	26,576	43,796
Total	64,230	31,041,787	836,379	318,740	517,639	836,490	318,785	517,705

Table 2: Statistics of our WDRASS dataset.

4 EXPERIMENTS

We assess the quality of our data comparing different retrieval and AS2-based QA systems trained with our data and the NQ dataset.

4.1 Experiment Setting

Evaluation Metrics. We measure the performance of retrieval results, i.e., a set of passages, and a QA pipeline output, i.e., top ranked selected answers, using the following metrics: Precision at 1 (P@1), i.e., the percentage of questions having relevant passage or correct answer sentence at top 1. Hit-rate-at-K (H@K), i.e., the percentage of questions having at least one relevant passage or one correct answer sentence in the first K items.

We measure retrieval in two settings: **Close**, which uses a search index of only 50 candidate passages for each question. **Open**, which, for each questions, uses an index constituted by the union of the 50 candidates of all test questions, i.e., $\sim 250k = \sim 5k \times 50$.

Models. We used three different models in cascade: (i) DPR for retrieving passages, which is based on the standard two-tower transformer; (ii) a passage reranker, which is trained with a cross-encoder transformer (i.e., question and passage are given in input together to the transformer); and (iii) a standard AS2 model, applied to all sentences of a passage for selecting the best answer sentence.

4.2 Dense Passage Retrieval Evaluation

In this experiment, we evaluate DPR encoders trained on NQ [16] and on our dataset WDRASS, using hit-rates at different Ks, derived from WDRASS, using both Close and Open settings.

The results are reported in Table 3. We noted that: First, DPR trained on WDRASS outperforms DPR trained on NQ, consistently. For example, with $K = 1$ (Hit-rate@K=P@K=Accuracy) DPR trained on WDRASS reaches 41.4%, which is ~ 10 absolute points better than DPR trained on NQ. In general, this is a reasonable result as the WDRASS test set is not a random sample from NQ data.

Second, there is a large difference between Close and Open setting. Specifically, to retrieve almost all the annotated relevant candidates of the dataset, it is required to set K to 5,000.

Finally, WDRASS outperforms NQ in both settings, the accuracy gap between the two datasets is larger for larger K , until $K = 10$, and then decreases (more slightly for the Open setting). For example, at $K = 10$, the gap is 17% and 18.2% under Close and Open settings, while at $K = 25$, it decreases to 12.3% and 17.4%, respectively.

In summary, models trained on WDRASS better capture question/passage relevancy (at least for the AS2 setting), e.g., with $K=50$ passages they reach a coverage of more than 90% vs. only 77% reached by models trained on NQ.

4.3 End-to-end Evaluation for ODQA-AS2

Finally, we compare different configurations of end-to-end QA pipelines for AS2, composed of DPR [12], passage reranking, and AS2 models. This end-to-end evaluation requires that the answers

Top-K	Close (1 50)		Open (1 250K)	
	NQ	WDRASS	NQ	WDRASS
1	0.318	0.414	0.243	0.327
2	0.395	0.531	0.307	0.424
3	0.446	0.600	0.342	0.485
4	0.491	0.659	0.375	0.532
5	0.540	0.701	0.408	0.568
10	0.664	0.834	0.498	0.680
25	0.828	0.951	0.626	0.800
50	1.000	1.000	0.777	0.903
500	-	-	0.897	0.960
5,000	-	-	0.971	0.991

Table 3: Hit-rate@K of DPR systems using encoders trained on NQ and WDRASS measured on Open and Close settings.

produced by each comparing system are manually judged by expert annotators. Given a question, we compare the following four configurations for selecting the top N relevant passages from an initial retrieval set of $K = 1000$ passages:

- C_0 : For each question, we select the top N passages from the K passages retrieved with DPR [12] trained on NQ dataset (we used the encoders available from github.com/facebookresearch/DPR).
- C_1 : For each question, we get the top K passages as we do for C_0 . Then, we rerank them with a Transformer-based reranker and select the top- N passages.
- C_2 : We use C_0 setup but using encoders trained on WDRASS.
- C_3 : We use the encoders of C_2 to select K passages and then we select N passages after applying a Transformer-based passage reranker also trained on WDRASS.

As the final evaluation is performed on answer sentences, for each of the configurations above, we split the N passages in sentences, and select the top M answer candidates from them, using the state-of-the-art AS2 model for WikiQA³. Table 4 shows the Hit-rate@M, (ranging from 1 to 5), averaged over a set of 1,296 questions. These are sampled from the Web and they are different from WDRASS. $S(N)$ is the number of sentences in the top N paragraphs, which of course is different for the different passage retrieval/ranking methods. We also present the break down of the performance by question type, i.e., factoid versus non-factoid.

The general finding is that training models on WDRASS consistently produces an improvement in answer Accuracy ($H@1=P@1$) over the baseline model trained on NQ data, i.e., 3.11%, 5.51%, and 6.83%, corresponding to WDRASS used for training only DPR, only reranker, and for both, respectively.

Finally, Table 5 shows an example of non-factoid questions (typically harder than factoid questions) with top answers produced by each configuration. We note that only C_3 , which exploits WDRASS for training DPR and passage reranker models, selects the correct answer sentence.

5 CONCLUSION

We have presented WDRASS, an AS2 dataset for retrieval and reranking. It consists of information collected in a vanilla end-to-end ODQA setting, including questions, retrieved documents, passages, and answer sentence candidates, manually labeled with

³Available at: github.com/alexa/wqa_tanda. This uses TANDA, which first fine-tunes RoBERTa on ASNQ and then on WikiQA.

Type	M	C ₀		C ₁		C ₂		C ₃	
		S(N)	H@M	S(N)	H@M	S(N)	H@M	S(N)	H@M
All	1	482	0.372	522	0.403	551	0.425	579	0.447
	2	588	0.454	621	0.479	650	0.502	675	0.521
	3	640	0.494	676	0.522	699	0.539	731	0.564
	4	667	0.515	700	0.540	724	0.559	761	0.587
	5	677	0.522	710	0.548	730	0.563	768	0.593
	Imp.	0.00%		3.09%		5.32%		7.48%	
	#Q	1,296							
Fact.	1	257	0.439	286	0.489	299	0.511	305	0.521
	2	301	0.515	330	0.564	336	0.574	347	0.593
	3	323	0.552	345	0.590	354	0.605	370	0.632
	4	343	0.586	359	0.614	361	0.617	376	0.643
	5	350	0.598	364	0.622	362	0.619	378	0.646
	Imp.	0.00%		4.96%		7.18%		8.21%	
	#Q	585							
N. F.	1	225	0.316	236	0.332	252	0.354	274	0.385
	2	287	0.404	291	0.409	314	0.442	328	0.461
	3	317	0.446	331	0.466	345	0.485	361	0.508
	4	324	0.456	341	0.480	363	0.511	385	0.541
	5	327	0.460	346	0.487	368	0.518	390	0.549
	Imp.	0.00%		1.55%		3.80%		6.89%	
	#Q	711							

Table 4: End-to-end human evaluation for ODQA AS2 setting, using an index constituted by ~130M passages.

q:	how do I change my clock forward an hour?
C_0 :	Refactoring is "the process of changing a software system in such a way that it does not alter the external behavior of the code yet improves its internal structure," according to Martin Fowler, the "father" of refactoring.
C_1 :	The modern day chronogrTaph works by pushing a start button, normally located at the two o'clock position, to begin recording time, and by pushing the same button to stop the recording.
C_2 :	The adjustment is made on New Year's Eve or the last day of June.
C_3 :	Change the time (or date) by pressing the button on the upper right side to advance the numbers or the button on the left side to move the numbers down.

Table 5: An example of non-factoid question with the top answer candidates selected from each retrieval configuration.

respect to answer sentence correctness. It accounts for ~800K annotated answer candidates responding to ~64K questions. The candidates are extracted from 90M documents, retrieved from a large index having 130M documents from CommonCrawl.

We show the effectiveness of our dataset by training and testing an end-to-end architecture for retrieval and reranking, based on Transformer models. We designed a passage retrieval setting that enables accurate retrieving of relevant passages to feed AS2 models in a standard retrieval-based ODQA. Our end-to-end system evaluation, validated by expert annotators, shows that models trained on WDRASS consistently outperform models trained on NQ (the largest and most accurate QA dataset, to our knowledge).

WDRASS is the first of its kind and can capture relevant data and signals for end-to-end ODQA. In addition, our work is also the first architectural proposal of open-domain AS2 using DPR. We believe our work, the dataset, the models, and the proposed ODQA system for AS2 can benefit several applications that require question-based accurate retrieval and reranking. We will make our models and data available to the research community to encourage future work.

REFERENCES

- [1] Qingyao Ai, Keping Bi, Jiafeng Guo, and W. Bruce Croft. 2018. Learning a Deep Listwise Context Model for Ranking Refinement. *CoRR* abs/1804.05936 (2018). arXiv:1804.05936 <http://arxiv.org/abs/1804.05936>
- [2] Weijie Bian, Si Li, Zhao Yang, Guang Chen, and Zhiqing Lin. 2017. A Compare-Aggregate Model with Dynamic-Clip Attention for Answer Selection. In *CIKM*. ACM, 1987–1990.
- [3] Daniele Bonadiman and Alessandro Moschitti. 2020. A Study on Efficiency, Accuracy and Document Structure for Answer Sentence Selection. *CoRR* abs/2003.02349 (2020). arXiv:2003.02349 <https://arxiv.org/abs/2003.02349>
- [4] Zhe Cao, Tao Qin, Tie-Yan Liu, Ming-Feng Tsai, and Hang Li. 2007. Learning to rank: from pairwise approach to listwise approach. In *Proceedings of the 24th international conference on Machine learning* (Corvallis, Oregon) (ICML '07). ACM, New York, NY, USA, 129–136. <https://doi.org/10.1145/1273496.1273513>
- [5] Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. 2017. Reading wikipedia to answer open-domain questions. *arXiv preprint arXiv:1704.00051* (2017).
- [6] Daniel Cohen, Liu Yang, and W Bruce Croft. 2018. WikiPassageQA: A benchmark collection for research on non-factoid answer passage retrieval. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*. 1165–1168.
- [7] Hengxin Fun, Sunil Gandhi, and Sujith Ravi. 2021. Efficient Retrieval Optimized Multi-task Learning. *arXiv preprint arXiv:2104.10129* (2021).
- [8] Siddhant Garg, Thuy Vu, and Alessandro Moschitti. 2020. TANDA: Transfer and Adapt Pre-Trained Transformer Models for Answer Sentence Selection. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*. AAAI Press, 7780–7788. <https://aaai.org/ojs/index.php/AAAI/article/view/6282>
- [9] Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. 2020. Realm: Retrieval-augmented language model pre-training. *arXiv preprint arXiv:2002.08909* (2020).
- [10] Gautier Izacard and Edouard Grave. 2020. Leveraging passage retrieval with generative models for open domain question answering. *arXiv preprint arXiv:2007.01282* (2020).
- [11] Mandar Joshi, Eunsol Choi, Daniel Weld, and Luke Zettlemoyer. 2017. TriviaQA: A Large Scale Distantly Supervised Challenge Dataset for Reading Comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, Vancouver, Canada, 1601–1611. <https://doi.org/10.18653/v1/P17-1147>
- [12] Vladimir Karpukhin, Barlas Oguz, Sewon Min, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense Passage Retrieval for Open-domain Question Answering. *arXiv preprint arXiv:2004.04906* (2020). <https://arxiv.org/abs/2004.04906>
- [13] Mostafa Keikha, Jae Hyun Park, and W. Bruce Croft. 2014. Evaluating answer passages using summarization measures. *Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval* (2014).
- [14] Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Matthew Kelcey, Jacob Devlin, Kenton Lee, Kristina N. Toutanova, Llion Jones, Ming-Wei Chang, Andrew Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. Natural Questions: a Benchmark for Question Answering Research. (2019).
- [15] Md Tahmid Rahman Laskar, Jimmy Xiangji Huang, and Enamul Hoque. 2020. Contextualized Embeddings based Transformer Encoder for Sentence Similarity Modeling in Answer Selection Task. In *Proceedings of the 12th Language Resources and Evaluation Conference*. European Language Resources Association, Marseille, France, 5505–5514. <https://www.aclweb.org/anthology/2020.lrec-1.676>
- [16] Kenton Lee, Ming-Wei Chang, and Kristina Toutanova. 2019. Latent Retrieval for Weakly Supervised Open Domain Question Answering. 6086–6096.
- [17] Patrick Lewis, Ethan Perez, Aleksandara Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *arXiv preprint arXiv:2005.11401* (2020).
- [18] Sewon Min, Jordan L. Boyd-Graber, Chris Alberti, Danqi Chen, Eunsol Choi, Michael Collins, Kelvin Guu, Hannaneh Hajishirzi, Kenton Lee, Jennimaria Palomaki, Colin Raffel, Adam Roberts, Tom Kwiatkowski, Patrick S. H. Lewis, Yuxiang Wu, Heinrich Küttler, Linqing Liu, Pasquale Minervini, Pontus Stenetorp, Sebastian Riedel, Sohee Yang, Minjoon Seo, Gautier Izacard, Fabio Petroni, Lucas Hosseini, Nicola De Cao, Edouard Grave, Ikuya Yamada, Sonse Shimaoka, Masatoshi Suzuki, Shumpei Miyawaki, Shun Sato, Ryo Takahashi, Jun Suzuki, Martin Fajcik, Martin Docekal, Karel Ondrej, Pavel Smrz, Hao Cheng, Yelong Shen, Xiaodong Liu, Pengcheng He, Weizhu Chen, Jianfeng Gao, Barlas Oguz, Xilun Chen, Vladimir Karpukhin, Stan Peshterliev, Dmytro Okhonko, Michael Sejr Schlichtkrull, Sonal Gupta, Yashar Mehdad, and Wen-tau Yih. 2021. NeurIPS 2020 EfficientQA Competition: Systems, Analyses and Lessons Learned. *CoRR* abs/2101.00133 (2021). arXiv:2101.00133 <https://arxiv.org/abs/2101.00133>
- [19] Jinfeng Rao, Hua He, and Jimmy J. Lin. 2016. Noise-Contrastive Estimation for Answer Selection with Deep Neural Networks. In *CIKM*. ACM, 1913–1916.
- [20] Revanth Gangi Reddy, Bhavani Iyer, Md. Arafat Sultan, Rong Zhang, Avi Sil, Vittorio Castelli, Radu Florian, and Salim Roukos. 2020. End-to-End QA on COVID-19: Domain Adaptation with Synthetic Training. *CoRR* abs/2012.01414 (2020). arXiv:2012.01414 <https://arxiv.org/abs/2012.01414>
- [21] A. Severyn and Alessandro Moschitti. 2015. Learning to Rank Short Text Pairs with Convolutional Deep Neural Networks. *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval* (2015).
- [22] Gehui Shen, Yunlun Yang, and Zhi-Hong Deng. 2017. Inter-Weighted Alignment Network for Sentence Pair Modeling. In *EMNLP'17*. Copenhagen, Denmark, 1179–1189. <https://doi.org/10.18653/v1/D17-1122>
- [23] Harish Tayyar Madabushi, Mark Lee, and John Barnden. 2018. Integrating Question Classification and Deep Learning for improved Answer Selection. In *COLING'18*. 3283–3294. <https://www.aclweb.org/anthology/C18-1278>
- [24] Ellen M Voorhees. 1999. The TREC-8 question answering track report. In *TREC*, Vol. 99. 77–82.
- [25] Mengqiu Wang, Noah A. Smith, and Teruko Mitamura. 2007. What is the Jeopardy Model? A Quasi-Synchronous Grammar for QA. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*. Association for Computational Linguistics, Prague, Czech Republic, 22–32. <https://aclanthology.org/D07-1003>
- [26] Yi Yang, Wen-tau Yih, and Christopher Meek. 2015. WikiQA: A Challenge Dataset for Open-Domain Question Answering. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Lisbon, Portugal, 2013–2018. <https://doi.org/10.18653/v1/D15-1237>
- [27] Seunghyun Yoon, Franck Dernoncourt, Doo Soon Kim, Trung Bui, and Kyomin Jung. 2019. A Compare-Aggregate Model with Latent Clustering for Answer Selection. *CoRR* abs/1905.12897 (2019). arXiv:1905.12897 <http://arxiv.org/abs/1905.12897>