MultiReQA: A Cross-Domain Evaluation for Retrieval Question Answering Models

Mandy Guo[a]*, **Yinfei Yang**[a]*, **Daniel Cer**[a], **Qinlan Shen**[b]†, **and Noah Constant**[a]

[a]Google Research
Mountain View, CA, USA

[b]Carnegie Mellon University
Pittsburgh, PA, USA

Abstract

Retrieval question answering (ReQA) is the task of retrieving a sentence-level answer to a question from an open corpus (Ahmad et al., 2019). This dataset paper presents MultiReQA, a new multi-domain ReQA evaluation suite composed of eight retrieval QA tasks drawn from publicly available QA datasets [1]. We explore systematic retrieval based evaluation and transfer learning across domains over these datasets using a number of strong baselines including two supervised neural models, based on fine-tuning BERT and USE-QA models respectively, as well as a surprisingly effective information retrieval baseline, BM25. Five of these tasks contain both training and test data, while three contain test data only. Performing cross training on the five tasks with training data shows that while a general model covering all domains is achievable, the best performance is often obtained by training exclusively on in-domain data.

1 Introduction

Retrieval-based question answering (QA) investigates the problem of finding answers to questions from an open corpus (Surdeanu et al., 2008; Yang et al., 2015; Chen et al., 2017; Lee et al., 2019; Ahmad et al., 2019; Chang et al., 2020; Ma et al., 2020). There is a growing interest in building scalable end-to-end question answering systems for large scale retrieval (Ahmad et al., 2019; Roy et al., 2020). Retrieval question answering (ReQA) (Ahmad et al., 2019), illustrated in Table 1, defines the task as *directly* retrieving an answer sentence from a corpus.[2] Motivated by real applications such as Google's Talk to Books [3], where sentence-level answers from books are retrieved to answer users' queries, ReQA is different from traditional machine reading for question answering or "reading comprehension" which aims to extract a short answer span from a given passage. Rather than just identifying answers within a short preselected passage that is provided to the model effectively by an oracle, retrieving sentence-level answers from a large pool of candidates directly addresses the real-world problem of searching for answers within a corpus. Sentences retrieved as answers in this manner can be used directly to answer questions. Alternatively, retrieved sentences, as well as possibly the passages that contains them, can be provided to a traditional Open Domain QA model (Chen et al., 2017; Karpukhin et al., 2020).

Question: In what year did Cortes send the first cochineal to Spain? **Answer in Context:** [...] It worked particularly well on silk, satin and other luxury textiles. **In 1523 Cortes sent the first shipment to Spain.** Soon cochineal began to arrive in European ports aboard convoys of Spanish galleons.

Table 1: ReQA example drawn from SQuAD. The goal is to retrieve the answer sentence (**bolded**) from an open corpus based on the meaning of the sentence and the surrounding context.

Recent research has shown promising results on developing neural models for retrieval tasks including ReQA, MS MARCO, and the retrieval part of open domain question QA (Roy et al., 2020; Karpukhin et al., 2020; Xiong et al., 2020; Luan et al., 2020). One challenge of employing neural models is that it usually requires a large amount of training data. While it is possible to get such data from a general domain, it may hard to get similar data for specialized domains, which is a common

* Corresponding authors: {xyguo, yinfeiy}@google.com

† Work done during an internship at Google Research.

[1] We released the sentence boundary annotation of MultiReQA: https://github.com/google-research-datasets/MultiReQA

[2] This can be contrasted to a two stage approach that first retrieves supporting text and then identifies the correct answer span (Chen et al., 2017; Lee et al., 2019)

[3] https://books.google.com/talktobooks/

Proceedings of the Second Workshop on Domain Adaptation for NLP, pages 94–104
April 20, 2021.

Dataset	Question	Answer
SearchQA	At age 33 in 1804, he started a new symphony, his 5th, with a Da-Da-Da-Duhg	This is the first movement of Beethoven's 5th symphony.
TriviaQA	From the Greek for color, what element, with an atomic number of 24, uses the symbol Cr?	Rubies and emeralds also owe their colors to chromium compounds.
HotpotQA	Lenny Young is a collaborator on the stop motion film released in what year?	Chicken Run is a 2000 stop-motion animated comedy film produced by the British studio Aardman Animations.
NQ	when was the last episode of vampire diaries aired	The series ran from September 10, 2009 to March 10, 2017 on The CW.
SQuAD	what decade did house music hit the mainstream in the us?	The early 1990s additionally saw the rise in mainstream US popularity for house music.
BioASQ	What chromosome is affected in Turner's syndrome?	The origin of sSMC of Turner syndrome with 45, X/46, X, + mar karyotype was almost all from sex chromosomes, and rarely from autosomes.
R.E.	Which year is Bird Girl and the Man Who Followed the Sun released?	Bird Girl and the Man Who Followed the Sun is a 1996 novel by Velma Wallis.
TextbookQA	which nervous system disease causes seizures?	Epilepsy is a disease that causes seizures.

Table 2: Example questions and answers from each dataset.

situation with examples including personal search over private repositories and search within enterprise environments (Hawking, 2004; Chirita et al., 2005). It is unknown how well a general domain model can perform on domain specific QA tasks or even the extent of transfer possible across different specialized domains.

In order to further investigate these questions within the context of the ReQA task, we propose a new common evaluation suite consisting of eight new datasets extracted from existing QA datasets. Five *in-domain* tasks include training and test data, while three *out-of-domain* tasks contain only test data. We provide cross domain baselines for neural and non-neural retrieval methods. Our baseline experiments use two competitive neural models, based on BERT (Devlin et al., 2019) and USE-QA (Yang et al., 2019), respectively, and BM25, a strong information retrieval baseline. BM25 performs surprisingly well on many retrieval question answering tasks, achieving the best performance on two of five in-domain tasks and all three out-of-domain tasks. Neural models achieve the highest performance on three of five in-domain tasks, outperforming BM25 by a wide margin on tasks with less token overlap between question and answer. Comparing general models trained on a mixture of QA training sets to specialized in-domain models trained on a single QA task reveals that models trained jointly on multiple datasets rarely outperform those trained on only in-domain data.

The main contribution of this paper is summarized as follows: 1) A new evaluation suit derived from existing QA tasks for measuring retrieval question answering performance in multiple domains; 2) Establish the strong baselines with key word based retrieval approach and neural retrieval models. 3) Exploring the domain transferability and limitation for existing retrieval models. Extensive experiments show that BM25 remains a strong baseline for all domains. While a general neural model covering all domains is achievable, the best performing neural model is often obtained by training exclusively on in-domain data.

2 Retrieval QA (ReQA)

ReQA formalizes the retrieval-based QA task as the identification of a sentence in-context that answers a provided question (Ahmad et al., 2019). Retrieval QA models are evaluated using Precision at 1 (P@1) and Mean Reciprocal Rank (MRR). The P@1 score tests whether the true answer sentence appears as the top-ranked candidate[4]. MRR, introduced for the evaluation of retrieval based QA systems (Voorhees, 2001; Radev et al., 2002), is calculated as MRR $= \frac{1}{N}\sum_{i=1}^{N}\frac{1}{rank_i}$, where N is the total number of questions, and $rank_i$ is the rank of the first correct answer for the ith question.

[4]Retrieval models are often measured by P@N (N=1,3,5,10). However, as our main concern is whether the question is correctly answered, we focus on P@1.

3 Multi-domain ReQA (MultiReQA)

The multi-domain ReQA (MultiReQA) test suite is composed of select datasets drawn from the MRQA shared task (Fisch et al., 2019a).[5] We follow the training, in-domain test, out-of-domain test splits defined in MRQA. The individual datasets are described below:

SearchQA Jeopardy question-answer pairs augmented with text snippets retrieved by Google (Dunn et al., 2017).

TriviaQA Trivia enthusiasts authored question-answer pairs. Answers are drawn from Wikipedia and Bing web search results, excluding trivia websites (Joshi et al., 2017b).

HotpotQA Wikipedia question-answer pairs. This dataset differs from the others in that the questions require reasoning over multiple supporting documents (Yang et al., 2018). The annotators generate the questions knowing the answers and the supporting contexts.

SQuAD 1.1 Wikipedia question-answer pairs (Rajpurkar et al., 2016a). Given the supporting contexts from Wikipedia, the annotators were asked to write questions such that the answers could be found in the contexts. Moreover, many of the questions are directly formed from parts of the supporting contexts.

NaturalQuestions (NQ) Questions are real queries issued by multiple users to Google search that retrieve a Wikipedia page in the top five search results. Answer text is drawn from the search results (Kwiatkowski et al., 2019). We removed the duplicate question-answer pairs in the in-domain test split, since during the original dataset construction, multiple raters were asked to select answers from the paragraphs. Unlike ReQA (Ahmad et al., 2019), we did not limit the questions and candidates to be only within the HTML paragraph block, and the candidates could contain lists and tables.

BioASQ Bio-medical question-answer pairs with answers annotated by domain experts and drawn from research articles (Tsatsaronis et al., 2015).

RelationExtraction (R.E.) Entity relation question-answer pairs, created by slot filling using the WikiReading dataset (Ahmad et al., 2019).

TextbookQA Multi-modal question-answer pairs taken from middle school science curricula (Kembhavi et al., 2017). In this paper, we only consider the text aspect of this task as defined by the original MRQA shared task.

Table 2 provides example question-answer sentence pairs. Datasets are converted from a span identification task to sentence-level retrieval. The questions from the original data are used without modification. Supporting documents are split into sentences using NLTK. [6] All resulting sentences become retrieval candidates. Answer spans identify sentences containing the correct answers. Spans covering multiple sentence are excluded.[7]

Table 3 provides statistics on the number of training set pairs and the number of questions, candidates and average answers per question in the evaluation data. Table 4 shows the average length of word tokens and degree of token overlap. SearchQA and HotpotQA have supporting documents split by [DOC]/[PAR] tags, so they have comparatively shorter context length. TriviaQA has much longer context length because all supporting documents were tokenized as one due to lack of clear division among special tags in the dataset. NaturalQuestions contain lists and tables that bring up the average answer length. SearchQA and SQuAD have high degree of question/answer overlap because the supporting documents in SearchQA are retrieved by search engine, and SQuAD questions are written with advance knowledge of the answers and supporting contexts. However, even though HotpotQA questions are also written with the knowledge of the answers and contexts, the degree of overlap is quite low likely due to the inclusion of multi-document inference.

[5] We exclude NewsQA, RACE, DROP, and DuoRC, as the majority of their questions are underspecified when taken out of their original context, making them inappropriate for a large-scale retrieval evaluations.

[6] As the datasets SearchQA, TriviaQA and HotpotQA contain special tags [DOC], [PAR], [SEP], and [TLE], we perform dataset-specific pre-processing to handle context splitting and tag removal. TriviaQA has [DOC] [TLE] [PAR] tags, but with no clear divisions to mark where the span of each kind of tags ends. We remove all the tags, and tokenize the article as if it does not have special tags. SearchQA uses [DOC] to separate the supporting snippets, [TLE] to mark the start of title, and [PAR] to mark start of the snippet content. We treat contents between two [DOC] tags as individual context. We then use NLTK to split the sentences within each context. The contents between [TLE] and [PAR] are used as a title feature. If the answer appears in the title feature, we do not add it as a positive answer. There are about 500 examples where the answer span is only in the title span, and we remove the corresponding questions. We follow the same procedure for HotpotQA, which uses [PAR] to separate supporting documents, and [SEP] to separate title and document content.

[7] This is typically due to sentence splitting errors by NLTK.

Dataset	Train	Test		
		Ques.	Cand.	Avg. ans. per ques.
SearchQA	629,160	16,476	454,836	5.47
TriviaQA	335,659	7,776	238,339	5.46
HotpotQA	104,973	5,859	52,191	1.69
SQuAD	87,133	10,485	10,642	1.09
NQ	106,521	4,131	22,118	1.06
BioASQ	-	1,503	14,158	2.85
R.E.	-	2,945	3,301	1.00
TextbookQA	-	1,497	3,701	3.31

Table 3: Statistics for each constructed dataset: # of training pairs, # of questions, # of candidates, and average # of answers per question.

Dataset	Question	Answer	Context
Average Length (Tokens)			
SearchQA	17.25	31.51	55.50
TriviaQA	15.56	33.88	747.75
HotpotQA	18.52	28.31	91.57
SQuAD	11.45	29.70	140.64
NQ	9.24	107.10	220.02
BioASQ	11.18	29.01	241.52
R.E.	9.15	27.51	29.14
TextbookQA	10.20	16.37	648.23
Question/Answer Token Overlap (%)			
SearchQA	-	37.83	55.23
TriviaQA	-	25.53	74.23
HotpotQA	-	29.08	49.16
SQuAD	-	43.03	56.36
NQ	-	23.50	36.87
BioASQ	-	23.08	53.40
R.E.	-	39.21	40.98
TextbookQA	-	25.64	82.54

Table 4: Average length (# of word tokens) and degree of question/answer token overlap of each constructed dataset.

4 Baseline Models

To establish strong baselines for the MultiReQA test suite, we use two neural models, based on BERT (Devlin et al., 2019) and USE-QA (Yang et al., 2019), respectively, as well as an well established term-based information retrieval baseline, BM25.

4.1 BERT

BERT dual encoders are used for retrieval tasks like translation retrieval (Feng et al., 2020) and QA passage retrieval (Roy et al., 2020; Karpukhin et al., 2020). We explore a BERT dual encoder as our first neural baseline, using the $\text{BERT}_{\text{BASE}}$ model,[8] due to memory constraints.[9]

Questions and answers are encoded using two separate towers with tied model weights. The question is fed into one tower and we take the embedding output of the CLS token as the question encoding. The answer text and context are concatenated as a long sequence, using segment IDs to separate them. The concatenated input is fed into the other tower. As with the question encoder, we take the CLS embedding as the answer encoding. To distinguish questions and answers, we add an additional *input type embedding* to each input token.[10] The final embeddings are l_2 normalized.

4.2 Universal Sentence Encoder QA

Following Ahmad et al. (2019), we employ the Universal Sentence Encoder QA (USE-QA) (Yang et al., 2019)[11] as another neural baseline. USE-QA is a multilingual QA retrieval model pretrained on billions of examples from web-crawled question answering corpora.[12] USE-QA encodes the question and answer separately using a transformer (Vaswani et al., 2017) based dual encoder architecture. The question embedding is obtained by average pooling over all token positions in the final transformer block follwed by fully-connected network. Answers and their context are encoded using a transformer for the answer text and a deep averaging network (DAN) (Iyyer et al., 2015) for context. Preliminary answer vectors are computed using average pooling over positions. The answer vector is then concatenated with the DAN based context vector and fed to a fully-connected network to compute the final joint representation.

4.3 BM25

Term frequency inverse document frequency (TF-IDF) based methods remain the dominant method for document retrieval, with the "Best Matching 25" (BM25) family of ranking functions providing a well established baseline (Robertson and Zaragoza, 2009). In previous work on open domain question answering, BM25 has been used to retrieve evi-

[8] The $\text{BERT}_{\text{BASE}}$ model uses 12 transformer layers with 12 attention heads, a hidden size of 768 and a filter size of 3072. The final embedding size is 768.

[9] We use in-batch negative sampling in the dual encoder training, which requires relatively large batch size. For more details of dual encoder training with negative sampling, see Gillick et al. (2018) and Guo et al. (2018).

[10] Note that we switch the final activation layer of the BERT CLS token from *tanh* to *gelu*.

[11] https://tfhub.dev/google/universal-sentence-encoder-multilingual-qa/1

[12] USE-QA uses a 6 layer transformer with 8 attention heads, a hidden size of 512 and a filter size of 2048. The context DAN encoder uses hidden sizes [320, 320, 512, 512] with residual connections. The feed-forward networks for question and answer both use hidden sizes [320, 512], so the final dimension of the encodings is 512.

dence text, and has been shown to be a particularly strong baseline on tasks where the question is written with advance knowledge of the answer (Lee et al., 2019).

The BM25 score of document D given query Q which contains words $q_1, ..., q_n$ is given by:

$$\sum_{i=1}^{n} \text{IDF}(q_i) \cdot \frac{f(q_i, D) \cdot (k_1 + 1)}{f(q_i, D) + k_1 \cdot (1 - b + b \cdot \frac{|D|}{\text{avgdl}})} \quad (1)$$

where $f(q_i, D)$ is q_i's term frequency in the document, $|D|$ is the length of the document in words, and $avgdl$ is the average document length across all documents. Scalars k_1 and b are free parameters. We concatenate the answer sentence and context as the document when applying BM25 for answer retrieval.[13]

5 Experiments

5.1 Fine-tuning and Configurations

BM25 We use the BM25 implementation in the Gensim library (Řehůřek and Sojka, 2010) with default k_1 and b settings. Inverse document frequency is calculated for each constructed dataset independently. We deploy two different tokenization methods for BM25: NLTK (Bird et al., 2009) and a WordPiece model (wpm) (Wu et al., 2016) following the BERT implementation.[14] The NLTK tokenizer does not normalize text, while the WordPiece model does by default. Our results in Table 5 for $\text{BM25}_{\text{word}}$ use NLTK without normalization, while BM25_{wpm} uses wpm with normalization.[15]

USE-QA The USE-QA model is already pre-trained specifically for retrieval QA tasks. We first evaluate the default model without any dataset specific fine-tuning. We further fine-tune USE-QA model with a discriminative ranking objective (Yang et al., 2019) on our training sets:[16]

$$P(a \mid q) = \frac{e^{\phi(q,a)}}{\sum_{\bar{a} \in \mathcal{A}} e^{\phi(q,\bar{a})}} \quad (2)$$

Above, q is the question, a is the correct answer, $\mathcal{A}$ is all answers in the same batch, which serve as sampled negatives, and $\phi(q, a)$ is the dot product of question and answer representations. We fine-tune USE-QA models on in-domain data for 10 epochs using batch size 64, and SGD with learning rate decaying exponentially from 0.01 to 0.001.

BERT Our BERT dual encoder is fine-tuned for retrieval with the same discriminative objective used for the USE-QA models.[17] We fine-tune for 10 epochs using batch size 128 and the default AdamW optimizer with learning rate 0.0001.[18]

5.2 Results

Table 5 shows baseline model performance on the MultiReQA evaluation suite for both precision at 1 (P@1) and Mean Reciprocal Rank (MRR). The highest score for each task is bolded. For P@1, the first two rows shows the results for $\text{BM25}_{\text{word}}$ and BM25_{wpm}. Notably, BM25_{wpm} performs better on 7 of 8 tasks, indicating that a careful selection of tokenization and normalization can improve the term-based model considerably. The advantage of BM25_{wpm} is particularly noticeable on datasets where the question is constructed without seeing the answer: SearchQA, TriviaQA, NQ, BioASQ and Relation Extraction. BM25_{wpm} also achieves the highest P@1 on 2 of 5 in-domain datasets and on all out-of-domain dataests.

The remaining rows show the results of the neural models: off-the-shelf USE-QA, fine-tuned versions of USE-QA and fine-tuned BERT dual encoders. We fine-tune on each in-domain dataset separately. The off-the-self USE-QA baseline is overall not competitive with BM25_{wpm}. However, when fine-tuned on in-domain data, USE-QA outperforms BM25_{wpm} on 3 of 5 in-domain datasets. Fine-tuned BERT often performs almost as well as fine-tuned USE-QA, suggesting there is only minimal benefit to QA specific pre-training.

The best neural models outperform BM25_{wpm} on Hotpot and NQ by +11.68% and +12.68% on P@1, respectively. This aligns with the statistics from Table 3, where token overlap between question and answer/context is low for these sets. BM25_{wpm} outperforms neural models, on datasets with higher token overlap between question and answer/context (e.g., SearchQA, R.E. and SQuAD w.r.t. all neural

[13] The answer sentence is included in the context, so it appears twice in the constructed documents. This allows multiple answers that share the same context to still receive unique scores.

[14] The wpm vocab is from $\text{BERT}_{\text{BASE}}$.

[15] We also experimented on SQuAD with removing normalization from wpm, and found that wpm still outperforms NLTK.

[16] Notably, this is the same discriminative objective used for the original USE-QA model

[17] Since BERT is originally pre-trained on masked language modeling and next-sentence prediction, fine-tuning is necessary to use it to perform retrieval tasks.

[18] For both USE-QA and BERT, hyper-parameters are tuned on a validation set (10%) split out from the training data.

Metric	Models	In-domain Datasets					Out-of-domain Datasets		
		SearchQA	TriviaQA	HotpotQA	NQ	SQuAD	BioASQ	R.E.	TextbookQA
P@1	$BM25_{word}$	30.94	39.35	21.04	10.07	61.50	6.38	55.75	8.39
	$BM25_{wpm}$	**35.86**	**43.26**	20.37	25.32	65.32	**8.31**	**64.04**	**8.52**
	USE-QA	31.17	28.60	18.12	24.71	51.02	5.58	52.05	7.52
	$USE\text{-}QA_{finetune}$	31.45	32.58	31.71	**38.00**	**66.83**	6.41	59.87	6.62
	$BERT_{finetune}$	30.20	29.11	**32.05**	36.22	55.13	5.71	49.89	6.29
MRR	$BM25_{word}$	47.75	51.58	33.07	15.51	69.16	10.37	71.27	17.23
	$BM25_{wpm}$	**52.25**	**55.80**	32.99	37.1	72.96	12.86	**79.86**	16.97
	USE-QA	47.52	40.26	22.65	34.73	62.08	12.31	67.41	16.92
	$USE\text{-}QA_{finetune}$	50.70	42.39	43.77	**52.27**	**75.86**	13.39	74.89	15.49
	$BERT_{finetune}$	47.08	41.34	**46.21**	52.02	64.74	**19.21**	65.21	**20.17**

Table 5: Precision at 1(P@1)(%) and Mean Reciprocal Rank (MRR)(%) on the constructed question answer retrieval datasets. $USE\text{-}QA_{finetune}$ and $BERT_{finetune}$ are fine-tuned on each in-domain dataset individually. The performance of fine-tuned models on out-of-domain datasets are the average score across all five fine-tuned models.

Metric	Test / Train	In-domain Datasets					Out-of-domain Datasets		
		SearchQA	TriviaQA	HotpotQA	NQ	SQuAD	BioASQ	R.E.	TextbookQA
P@1	SearchQA	31.45	35.48	16.04	24.69	46.60	6.52	60.03	6.66
	TriviaQA	28.44	32.58	14.91	22.58	38.87	4.45	60.84	4.06
	HotpotQA	30.79	32.70	**31.71**	26.45	56.17	5.65	57.21	6.52
	NQ	28.80	31.77	17.64	**38.00**	52.23	6.52	55.48	7.66
	SQuAD	31.44	35.21	20.25	28.32	**66.83**	**7.65**	**63.73**	8.32
	Joint	**32.24**	37.40	26.54	36.35	60.81	7.58	62.71	7.52
	$Joint_{No\ TriviaQA}$	31.92	**37.71**	29.68	36.23	64.00	6.78	61.69	**8.72**
MRR	SearchQA	50.70	47.88	25.88	36.31	57.83	13.34	75.51	15.19
	TriviaQA	44.57	42.39	23.40	32.77	47.50	9.26	75.88	10.49
	HotpotQA	47.17	44.41	**43.77**	36.99	66.25	32.15	72.54	15.08
	NQ	45.08	44.39	26.57	**52.27**	62.88	13.77	70.07	17.71
	SQuAD	48.70	48.16	30.12	38.79	**75.86**	**15.75**	**78.50**	**18.71**
	Joint	**51.04**	**50.88**	38.95	50.11	71.02	14.86	78.05	16.61
	$Joint_{No\ TriviaQA}$	50.80	50.77	41.62	49.93	73.71	14.69	77.04	18.64

Table 6: P@1(%) and MRR(%) of USE-QA models fine-tuned on either one or all in-domain datasets, evaluated across all datasets. **Joint**: Fine-tune on all in-domain datasets together. $\mathbf{Joint_{No\ TriviaQA}}$: Same as "Joint", but removing TriviaQA from the fine-tuning data pool.

models except $USE\text{-}QA_{finetune}$) and paradoxically the particularly difficult TriviaQA task.

A very similar pattern of results is seen for MRR, with the exception that $BERT_{finetune}$ performs best on BioASQ and TextbookQA. We observe that the vocabulary of BioASQ and TextbookQA are different from the other datasets, including more specialized technical terms. Superior MRR performance could be due to better representations of novel words, computed from the composition of sub-word tokens.[19] However, it's not clear why $BM25^{wpm}$, also using sub-word tokenization, performs best on these datasets for P@1.

5.3 Transfer Learning across Domains

The previous section shows that the strongest baselines are the USE-QA and BERT models, fine-tuned on in-domain data, with USE-QA slightly outperforming BERT. In order to better understand generalization across QA tasks, we experiment with training and evaluating on different dataset pairings, focusing on the USE-QA model. Table 6 shows the performance of models trained on each individual dataset, as well as a model trained jointly on all available in-domain datasets.

Each column compares performance of different models on a specific test set. The best numbers for each test set are bolded. In general, models trained on an individual dataset achieve the best (or near-best) performance on their associated evaluation set. TriviaQA is an exception, performing poorly on its own evaluation data and nearly all other datasets. This suggests training on the TriviaQA sentence-level retrieval task is more difficult than other datasets. Critically, TriviaQA requires reasoning across multiple sources of evidence (Joshi et al., 2017a), with the meaning of complete sentences annotated with answer spans often not directly answering their associate questions.

[19] $BM23_{wpm}$ benefits from subword tokens but lacks the ability to understand how adjacent sub-word tokens compose a larger meaningful unit.

Joint models use combined training sets. The model denoted as Joint trains on all the datasets. $\text{Joint}_{\text{No TriviaQA}}$ trains on all datasets except TrivialQA, motivated by the poor performance of models trained on only TriviaQA data. The model trained over all available data is competitive, but the performance on some datasets, e.g. NQ and SQuAD, is significantly lower than the individually-trained models. By removing TriviaQA, the combined model gets close to the individual model performance on NQ and SQuAD, and achieves the best P@1 performance on TriviaQA and TextbookQA.

6 Analysis

6.1 Does Context Help?

Candidate answers may be not fully interpretable when taken out of their surrounding context (Ahmad et al., 2019). In this section we investigate how model performance changes when removing context. We experiment with one BM25 model and one neural model, by picking the best performing models from previous experiments: BM25_{wpm} and $\text{USE-QA}_{\text{finetune}}$. Recall, $\text{USE-QA}_{\text{finetune}}$ models are fine-tuned on each individual dataset.

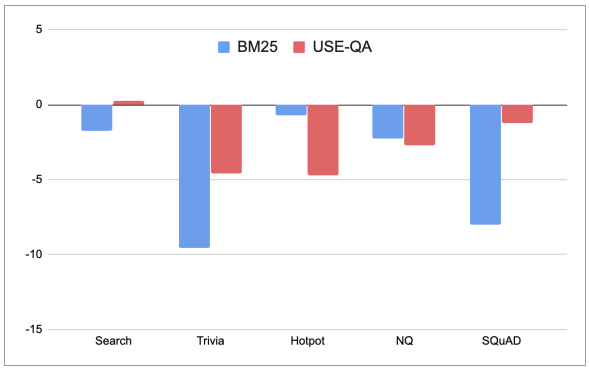

Figure 1: Performance change on P@1(%) for BM25_{wpm} and $\text{USE-QA}_{\text{finetune}}$ when candidate selection is blind to answer context.

Figure 1 illustrates the change in performance when models are restricted to evaluating candidate answers without context.[20] For the USE-QA model, the performance drop by excluding answer context is less than 5% on all datasets. The drop in BM25 performance is larger, supporting the hypothesis that BM25's token overlap heuristic is effective over large spans of text, while the neural model obtains a "deeper" semantic understanding and thus extracts more signal out of a single sentence.

[20] We report P@1 here, but observed similar trends in MRR.

6.2 Error Analysis

In this section we examine some typical failure cases of the BM25_{wpm} and $\text{USE-QA}_{\text{finetune}}$ models. As a first observation, the two models retrieve very different answers. For example, we find that on Natural Questions, the two models' top-ranked answers disagree on 64.75% questions.[21] The other datasets have similar levels of disagreement. This suggests that the models have different strengths, and that a combination of these modeling techniques could leads to a significant improvement.

Table 7 shows examples where the models retrieve different answers, and both are incorrect. In the first example, the BM25_{wpm} retrieves the correct context by matching the keyword "Salton Sea". But it fails to retrieve the correct sentence, as none of the keywords in the question appear in the target answer. On the other hand, the $\text{USE-QA}_{\text{finetune}}$ model understands the question is asking about some sort of animal living in the sea, but fails to connect to the Salton Sea specifically. Similarly, in the second example, both models retrieve sentences that match some keywords from the question. The BM25_{wpm} matches keywords "Spencer" and "Maine", but misses that the question is looking for an invention. The $\text{USE-QA}_{\text{finetune}}$ matches "Spencer", and is able to connect "invent" with "discover", but surfaces the wrong discovery. Overall, we observe term based models very often retrieve the correct context, but then fail to identify the correct sentence as the answer. Conversely, neural models seems to better understand the question, but sometimes fails to recognize important keywords.

7 Related Work

Open domain QA involves finding answers to questions within large document collections (Voorhees and Tice, 2000). The ground-truth answer for many evaluations is a span often containing a word or a short phrase (i.a., Kwiatkowski et al. (2019); Chen et al. (2017); Rajpurkar et al. (2016b)).

Karpukhin et al. (2020) and Xiong et al. (2020) explored passage level retrieval for QA. Seo et al. (2018) constructs a phrase-indexed QA challenge benchmark retrieving phrases, allowing for a direct F_1 and exact-match evaluation on SQuAD. (Seo et al., 2019) demonstrates phrase-indexed QA systems can be built using a combination of dense (neural) and sparse (term-frequency based)

[21] Note that even if the models retrieve different answers, both answers could still be correct.

Example 1 (from NQ): what kind of fish live in the salton sea

Correct Answer: [...] Due to the high salinity , very few fish species can tolerate living in the Salton Sea . *Introduced tilapia are the main fish that can tolerate the high salinity levels and pollution* . Other freshwater fish species live in the rivers and canals that feed the Salton Sea , including threadfin shad . [...]

USE-QA$_{\text{finetune}}$: [...] It may also drift in to the south - western part of the Baltic Sea (where it can not breed due to the low salinity) . *Similar jellyfish – which may be the same species – are known to inhabit seas near Australia and New Zealand* . The largest recorded specimen found washed up on the shore of Massachusetts Bay in 1870 . [...]

BM25$_{\text{wpm}}$: [...] Introduced tilapia are the main fish that can tolerate the high salinity levels and pollution . *Other freshwater fish species live in the rivers and canals that feed the Salton Sea , including threadfin shad , carp, red shiner , channel catfish , white catfish , largemouth bass , mosquitofish , sailfin molly , and the vulnerable desert pupfish* . [...]

Example 2 (from TriviaQA): What was invented in the 1940s by Percy Spencer, an American self-taught engineer from Howland, Maine, who was building magnetrons for radar sets?

Correct Answer: [...] After experimenting, he realized that microwaves would cook foods quickly - even faster than conventional ovens that cook with heat. *The Raytheon Corporation produced the first commercial microwave oven in 1954; it was called the 1161 Radarange.* It was large, expensive, and had a power of 1600 watts. [...]

USE-QA$_{\text{finetune}}$: [...] Because of his accomplishments, Spencer was awarded the Distinguished Service Medal by the U.S. Navy and has a building named after him at Raytheon. *Percy Spencer, while working for the Raytheon Company, discovered a more efficient way to manufacture magnetrons.* In 1941, magnetrons were being produced at a rate of 17 per day. [...]

BM25$_{\text{wpm}}$: [...] By the end of 1971, the price of countertop units began to decrease and their capabilities were expanded. *Spencer, born in Howland, Maine, was orphaned at a young age.* Although he never graduated from grammar school, he became Senior Vice President and a member of the Board of Directors at Raytheon, receiving 150 patents during his career [...]

Table 7: Examples where both the BM25$_{\text{wpm}}$ and USE-QA$_{\text{finetune}}$ models get wrong. *Italics* indicate the answer sentence. At most one sentence before/after the answer is shown, although the original context may be longer.

indices. Roy et al. (2020) investigates the retrieval of sentence-level answers from a language agnostic candidate pool. Chang et al. (2020) investigates the pre-training tasks for retrieving answers from a large scale candidate pool.

Surdeanu et al. (2008) provides a dataset consisting of 142,627 question-answer pairs from Yahoo! Answers "how to" questions, with the goal of retrieving the correct answer to a given question from the set of all answers. WikiQA (Yang et al., 2015) is another sentence-level answer selection dataset consisting of 3,047 questions and 29,258 candidate answers, split into train, dev, and test. These datasets, however, are either limited to a specific type of question, or limited to a small set of candidates. we propose a more comprehensive eval covering multiple domains and include tasks at a much larger scale. Additionally, folding the various MRQA in-domain and out-of-domain datasets into a single eval allows us to directly investigate cross-domain generalization.

8 Conclusion

In this paper, we convert eight existing QA tasks from the MRQA shared task (Fisch et al., 2019b) into sentence-level retrieval tasks, by treating the sentence containing the ground-truth span as the target sentence-level answer. In additional to a new evaluation suite for sentence level retrieval, we provide strong baselines using unsupervised term-based information retrieval methods (BM25), and three neural models, off-the-self USE-QA, fine-tuned USE-QA, and BERT dual encoders.

Overall, BM25's classical term-based retrieval approach is a surprisingly strong baseline, and one that could likely be improved further using additional information retrieval techniques such as normalization and synonym matching. The neural models, however, can be trained end-to-end without feature engineering, and perform particularly well on tasks with a low degree of question/answer token overlap, or in situations where context is limited. The neural model performance can also be improved through the addition of in-domain training data. However, we find that QA tasks are not all alike and having training data in the precise target domain is important.

References

Amin Ahmad, Noah Constant, Yinfei Yang, and Daniel Cer. 2019. ReQA: An evaluation for end-to-end answer retrieval models. In *Proceedings of the 2nd Workshop on Machine Reading for Question Answer-*

ing, pages 137–146, Hong Kong, China. Association for Computational Linguistics.

Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural language processing with Python: analyzing text with the natural language toolkit*. " O'Reilly Media, Inc.".

Wei-Cheng Chang, Felix X. Yu, Yin-Wen Chang, Yiming Yang, and Sanjiv Kumar. 2020. Pre-training tasks for embedding-based large-scale retrieval. In *International Conference on Learning Representations*.

Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. 2017. Reading Wikipedia to answer open-domain questions. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1870–1879, Vancouver, Canada. Association for Computational Linguistics.

Paul Alexandru Chirita, Wolfgang Nejdl, Raluca Paiu, and Christian Kohlschütter. 2005. Using odp metadata to personalize search. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 178–185.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Matthew Dunn, Levent Sagun, Mike Higgins, V. Ugur Güney, Volkan Cirik, and Kyunghyun Cho. 2017. Searchqa: A new q&a dataset augmented with context from a search engine. *CoRR*, abs/1704.05179.

Fangxiaoyu Feng, Yinfei Yang, Daniel Cer, Naveen Arivazhagan, and Wei Wang. 2020. Language-agnostic bert sentence embedding.

Adam Fisch, Alon Talmor, Robin Jia, Minjoon Seo, Eunsol Choi, and Danqi Chen. 2019a. MRQA 2019 shared task: Evaluating generalization in reading comprehension. In *Proceedings of the 2nd Workshop on Machine Reading for Question Answering*, pages 1–13, Hong Kong, China. Association for Computational Linguistics.

Adam Fisch, Alon Talmor, Robin Jia, Minjoon Seo, Eunsol Choi, and Danqi Chen. 2019b. MRQA 2019 shared task: Evaluating generalization in reading comprehension. In *Proceedings of 2nd Machine Reading for Reading Comprehension (MRQA) Workshop at EMNLP*.

Daniel Gillick, Alessandro Presta, and Gaurav Singh Tomar. 2018. End-to-end retrieval in continuous space. *arXiv preprint arXiv:1811.08008*.

Mandy Guo, Qinlan Shen, Yinfei Yang, Heming Ge, Daniel Cer, Gustavo Hernandez Abrego, Keith Stevens, Noah Constant, Yun-Hsuan Sung, Brian Strope, and Ray Kurzweil. 2018. Effective parallel corpus mining using bilingual sentence embeddings. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 165–176, Belgium, Brussels. Association for Computational Linguistics.

David Hawking. 2004. Challenges in enterprise search. In *ADC*, volume 4, pages 15–24. Citeseer.

Mohit Iyyer, Varun Manjunatha, Jordan Boyd-Graber, and Hal Daumé III. 2015. Deep unordered composition rivals syntactic methods for text classification. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1681–1691, Beijing, China. Association for Computational Linguistics.

Mandar Joshi, Eunsol Choi, Daniel Weld, and Luke Zettlemoyer. 2017a. 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)*, pages 1601–1611, Vancouver, Canada. Association for Computational Linguistics.

Mandar Joshi, Eunsol Choi, Daniel S. Weld, and Luke Zettlemoyer. 2017b. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, Vancouver, Canada. Association for Computational Linguistics.

Vladimir Karpukhin, Barlas Oğuz, Sewon Min, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen tau Yih. 2020. Dense passage retrieval for open-domain question answering.

Aniruddha Kembhavi, Minjoon Seo, Dustin Schwenk, Jonghyun Choi, Ali Farhadi, and Hannaneh Hajishirzi. 2017. Are you smarter than a sixth grader? textbook question answering for multimodal machine comprehension. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

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. *Transactions of the Association of Computational Linguistics*.

Kenton Lee, Ming-Wei Chang, and Kristina Toutanova. 2019. Latent retrieval for weakly supervised open domain question answering. In *Proceedings of the*

57th Annual Meeting of the Association for Computational Linguistics, pages 6086–6096, Florence, Italy. Association for Computational Linguistics.

Yi Luan, Jacob Eisenstein, Kristina Toutanova, and Michael Collins. 2020. Sparse, dense, and attentional representations for text retrieval.

Ji Ma, Ivan Korotkov, Yinfei Yang, Keith Hall, and Ryan McDonald. 2020. Zero-shot neural retrieval via domain-targeted synthetic query generation.

Dragomir R. Radev, Hong Qi, Harris Wu, and Weiguo Fan. 2002. Evaluating web-based question answering systems. In *Proceedings of the Third International Conference on Language Resources and Evaluation (LREC'02)*, Las Palmas, Canary Islands - Spain. European Language Resources Association (ELRA).

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016a. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016b. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.

Radim Řehůřek and Petr Sojka. 2010. Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pages 45–50, Valletta, Malta. ELRA. http://is.muni.cz/publication/884893/en.

Stephen Robertson and Hugo Zaragoza. 2009. The probabilistic relevance framework: Bm25 and beyond. *Found. Trends Inf. Retr.*, 3(4):333–389.

Uma Roy, Noah Constant, Rami Al-Rfou, Aditya Barua, Aaron Phillips, and Yinfei Yang. 2020. Lareqa: Language-agnostic answer retrieval from a multilingual pool. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Minjoon Seo, Tom Kwiatkowski, Ankur Parikh, Ali Farhadi, and Hannaneh Hajishirzi. 2018. Phrase-indexed question answering: A new challenge for scalable document comprehension. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 559–564, Brussels, Belgium. Association for Computational Linguistics.

Minjoon Seo, Jinhyuk Lee, Tom Kwiatkowski, Ankur Parikh, Ali Farhadi, and Hannaneh Hajishirzi. 2019. Real-time open-domain question answering with dense-sparse phrase index. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4430–4441, Florence, Italy. Association for Computational Linguistics.

Mihai Surdeanu, Massimiliano Ciaramita, and Hugo Zaragoza. 2008. Learning to rank answers on large online QA collections. In *Proceedings of ACL-08: HLT*, pages 719–727, Columbus, Ohio. Association for Computational Linguistics.

George Tsatsaronis, Georgios Balikas, Prodromos Malakasiotis, Ioannis Partalas, Matthias Zschunke, Michael R. Alvers, Dirk Weissenborn, Anastasia Krithara, Sergios Petridis, Dimitris Polychronopoulos, Yannis Almirantis, John Pavlopoulos, Nicolas Baskiotis, Patrick Gallinari, Thierry Artières, Axel-Cyrille Ngonga Ngomo, Norman Heino, Éric Gaussier, Liliana Barrio-Alvers, Michael Schroeder, Ion Androutsopoulos, and Georgios Paliouras. 2015. An overview of the bioasq large-scale biomedical semantic indexing and question answering competition. In *BMC Bioinformatics*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 6000–6010. Curran Associates, Inc.

Ellen M. Voorhees. 2001. The trec question answering track. *Nat. Lang. Eng.*, 7(4):361–378.

Ellen M. Voorhees and Dawn M. Tice. 2000. Building a question answering test collection. In *Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '00, pages 200–207, New York, NY, USA. ACM.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*.

Lee Xiong, Chenyan Xiong, Ye Li, Kwok-Fung Tang, Jialin Liu, Paul Bennett, Junaid Ahmed, and Arnold Overwijk. 2020. Approximate nearest neighbor negative contrastive learning for dense text retrieval.

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*, pages 2013–2018, Lisbon, Portugal. Association for Computational Linguistics.

Yinfei Yang, Daniel Cer, Amin Ahmad, Mandy Guo, Jax Law, Noah Constant, Gustavo Hernandez Abrego, Steve Yuan, Chris Tar, Yun-Hsuan

Sung, et al. 2019. Multilingual universal sentence encoder for semantic retrieval. *arXiv preprint arXiv:1907.04307*.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W. Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. 2018. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*.