Distant Supervision for Multi-Stage Fine-Tuning in Retrieval-Based Question Answering

Yuqing Xie,[1,2] Wei Yang,[1,2] Luchen Tan,[1,2] Kun Xiong,[2] Nicholas Jing Yuan,[3]
Baoxing Huai,[3] Ming Li,[1,2] Jimmy Lin[1,2]*

[1] David R. Cheriton School of Computer Science, University of Waterloo
[2] RSVP.ai [3] Huawei Cloud & AI

ABSTRACT

We tackle the problem of question answering directly on a large document collection, combining simple "bag of words" passage retrieval with a BERT-based reader for extracting answer spans. In the context of this architecture, we present a data augmentation technique using distant supervision to automatically annotate paragraphs as either positive or negative examples to supplement existing training data, which are then used together to fine-tune BERT. We explore a number of details that are critical to achieving high accuracy in this setup: the proper sequencing of different datasets during fine-tuning, the balance between "difficult" vs. "easy" examples, and different approaches to gathering negative examples. Experimental results show that, with the appropriate settings, we can achieve large gains in effectiveness on two English and two Chinese QA datasets. We are able to achieve results at or near the state of the art without any modeling advances, which once again affirms the cliché "there's no data like more data".

CCS CONCEPTS

• **Information systems → Question answering**.

KEYWORDS

data augmentation, reranking, BERT

ACM Reference Format:
Yuqing Xie,[1,2] Wei Yang,[1,2] Luchen Tan,[1,2] Kun Xiong,[2] Nicholas Jing Yuan,[3] Baoxing Huai,[3] Ming Li,[1,2] Jimmy Lin[1,2] . 2020. Distant Supervision for Multi-Stage Fine-Tuning in Retrieval-Based Question Answering. In *Proceedings of The Web Conference 2020 (WWW '20), April 20–24, 2020, Taipei, Taiwan.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3366423.3380060

1 INTRODUCTION

BERT [6] exemplifies a large family of deep neural models that take advantage of massive pretraining on language modeling tasks [16, 17]. With these models, researchers have demonstrated impressive gains in a broad range of NLP tasks, from sentence classification to paraphrase detection to sequence labeling.

Recently, Yang et al. [29] showed that combining a BERT-based reader with passage retrieval using BM25 in a system called BERTserini yields a large improvement in question answering accuracy in a retrieval-based setting, identifying answers from a Wikipedia corpus [4]. BERTserini adopts a simple architecture and a straightforward method to combine BERT with off-the-shelf retrieval techniques, and has become a reference point for researchers working on this problem [1, 7, 8, 13, 20, 26]. Our paper builds on BERTserini's basic design and explores how much further we can improve effectiveness by data augmentation *alone*. We take advantage of distant supervision techniques to gather "free" training data to fine-tune BERT. Experiments show that, using the same reader model as Yang et al. [29], our data augmentation techniques yield additional large improvements in standard metrics. To illustrate the robustness of our methods, we also demonstrate gains on another English QA dataset and present results for two Chinese QA datasets that have not to date been evaluated in a retrieval-based setting.

While data augmentation using distant supervision has, of course, been studied by many researchers in the past, our work focuses specifically on two under-explored aspects of retrieval-based question answering. In particular, we make two contributions to the understanding of distant supervision techniques in this context:

- First, most previous work on distant supervision focuses on generating positive examples. However, in a retrieval-based setting where the reader consumes the output of passage retrieval, the model will encounter many non-relevant passages, which means that a data collection strategy focused only on positive examples will be inadequate. We show that using existing datasets to identify negative training examples is critical to effectiveness, and explore different strategies for gathering negative examples.
- Second, the literature provides little guidance on how to integrate existing training data with data gathered via distant supervision in the context of fine-tuning BERT for ranking. We show that a naïve strategy of simply combining all data may not be the best method. Instead, we propose a stage-wise approach to fine-tuning BERT with heterogeneous data, beginning with the dataset that is "furthest" from the test data and ending with the "closest".

Combining our innovations, we are able to achieve effectiveness at or near the state of the art on four question answering datasets. One notable strength of our approach is that our results are obtained *without any model changes*, which once again affirms the cliché "there's no data like more data". Since data augmentation is orthogonal to model improvements, our techniques can be applied to future modeling advances.

*The first two authors contributed equally to this research.

WWW '20, April 20–24, 2020, Taipei, Taiwan

ACM ISBN 978-1-4503-7023-3/20/04.
https://doi.org/10.1145/3366423.3380060

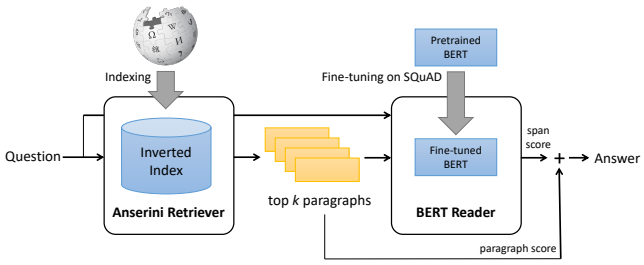

Figure 1: Our two-stage architecture for retrieval-based question answering, the same design as BERTserini [29].

2 RELATED WORK AND METHODS

Our work tackles the retrieval-based variant of the question answering problem, where the system is provided with a large corpus. This stands in contrast to reading comprehension datasets such as SQuAD [18], where the system works with a single pre-determined document, or most QA benchmarks today such as TrecQA [32], WikiQA [31], and MS MARCO passage reranking [2], where the system is provided a list of candidate passages to choose from. Our task definition, which combines a strong element of information retrieval, traces back to the Text Retrieval Conferences (TRECs) in the late 1990s [23], but there is a recent resurgence of interest in this formulation, starting with Chen et al. [4]. Following this work, we have seen many papers on retrieval-based question answering, and BERTserini is a typical two-stage pipeline that combines keyword-based retrieval with BERT-based (re)ranking.

2.1 Basic Setup

In this work, we fix the underlying model and focus on data augmentation techniques to explore how to best fine-tune BERT in a retrieval-based multi-stage question answering pipeline. Following BERTserini [29], we first use passage retrieval to identify relevant paragraphs from Wikipedia using the Anserini IR toolkit [27, 28] and then pass the paragraphs to a BERT reader for answer span extraction (see Figure 1); hence the name of the system.

We adopted the "paragraph" setup in BERTserini [29]: the input corpus, a Wikipedia dump, is pre-segmented into paragraphs at indexing time, each of which is treated as a "document" for retrieval purposes. The question is used as a "bag of words" query to retrieve the top k candidate paragraphs using BM25 ranking.

During the inference stage, the retrieved paragraphs are fed into the fine-tuned BERT reader, along with the original natural language question. For each candidate paragraph, the reader selects the best text span and provides a score. We also follow Yang et al. [29] and remove the final softmax layer over different answer spans to make the reader score comparable across the candidate paragraphs. Finally, an aggregator combines the reader scores with the BM25 scores via linear interpolation:

$$S = (1 - \mu) \cdot S_{\text{BM25}} + \mu \cdot S_{\text{BERT}} \tag{1}$$

where $\mu \in [0, 1]$ is a hyperparameter to be tuned.

2.2 Distant Supervision

The roots of the distant supervision techniques we use trace back to at least the 1990s [19, 33], although the term had not yet been coined. Such techniques have recently become commonplace, especially as a way to gather large amounts of labeled examples for data-hungry neural networks and other machine learning algorithms. Specific recent applications in question answering include Bordes et al. [3], Chen et al. [4], Lin et al. [14], as well as Joshi et al. [9] for building benchmark test collections. However, as we explain below, our approach aims to address a number of under-explored issues that are specific to the retrieval-based setting.

One main shortcoming of BERTserini as implemented by Yang et al. [29] was that they only fine-tune BERT on the original SQuAD dataset, which means that the BERT reader is exposed to an impoverished set of examples; all SQuAD data come from a total of only 442 documents. This contrasts with the diversity of paragraphs that the model will likely encounter at inference time in the retrieval-based setting, since they are selected from potentially millions of articles. Also, the output of passage retrieval may contain many negative examples (i.e., paragraphs that don't contain answers), which does not accurately match the prevalence of answers in the original training data. Additionally, when performing QA on Wikipedia using a machine-reading comprehension dataset, we do not have labels for most paragraphs (since they are not from the original source documents). The solution, of course, is to fine-tune BERT with labeled paragraphs of the type that it is likely to encounter at inference time. Distant supervision can provide a bridge.

Starting from a source dataset comprising question–answer pairs (for sample, SQuAD), we can create additional training examples by fetching paragraphs from the corpus using the system's own passage retrieval algorithm (with the question as the query) and automatically annotate these paragraphs based on the ground truth answers provided in the source dataset. Using the system's own passage retrieval algorithm ensures that the model is exposed to the types of input it will receive at inference time.

We denote a paragraph as a positive example if the ground truth answer appears in it; otherwise we consider it a negative example. We keep the best positive example for each question (i.e., the highest BM25 score), and examined three different methods for selecting negative examples:

- Top-down: We choose negative examples with the highest paragraph scores from the retrieved paragraphs.
- Bottom-up: We choose negative examples with the lowest paragraph scores from the retrieved paragraphs.
- Random: We randomly sample negative examples from the retrieved paragraphs.

Our intuition is that top-down sampling selects "hard" examples, since the paragraphs have high BM25 scores, while bottom-up sampling selects "easy" examples. Random sampling attempts to obtain diversity.

One important hyperparameter is the number of negative examples to select for each positive example; ideally, this distribution should match the actual prevalence of correct answers that the reader encounters at test time, giving the model an accurate prior for answer correctness.

Another important design decision is how to make use of both the source data (denoted SRC) and the distantly-supervised data (denoted DS). We refer to the training set of only positive augmented examples as DS(+) and refer to the training set containing both positive and negative examples as DS(±). Other than different datasets we can use, there are three possibilities when considering the fine-tuning order:

- SRC + DS: Fine-tune BERT with all data, "lumped" together as a single, larger training set. In practice, this means that the source and augmented data are shuffled together.
- DS → SRC: Fine-tune the reader in stages, first on the augmented data and then on the source dataset.
- SRC → DS: Fine-tune the reader in stages, first on the source dataset and then on the augmented data.

Our experiments explore the facets of distant supervision described above: different strategies for gathering negative examples and the proper sequencing of different datasets during fine-tuning. Results show that these choices have a large impact on effectiveness.

3 EXPERIMENTAL SETUP

To demonstrate the generalizability of our data augmentation techniques, we conducted experiments on two English datasets: SQuAD (v1.1) and TriviaQA [9] (the unfiltered version). The 2016/12/21 dump of English Wikipedia provides the document collection from which we are retrieving answers, following Chen et al. [4]. We also examined two Chinese machine reading comprehension datasets: CMRC [5] and DRCD [21]. CMRC is in simplified Chinese while DRCD is in traditional Chinese. For Chinese, we used the 2018/12/01 dump of Chinese Wikipedia, tokenized with Lucene's `CJKAnalyzer` into overlapping bigrams. We applied `hanziconv`[1] to transform the corpus into simplified characters for CMRC and traditional characters for DRCD.

Following Yang et al. [29], to evaluate under a retrieval-based setting, we simply disregard the ground truth paragraphs from the original datasets and consider only the exact answer strings. As in previous work, we use the exact match (EM) score and F_1 score (at the token level) as evaluation metrics. In addition, to examine the quality of retrieved paragraphs, we compute recall (R), the fraction of questions for which the correct answer appears in *any* retrieved paragraph. Note that this recall is *not* the same as the token-level recall component in the F_1 score. To make our results comparable to BERTserini, we also use Anserini [27, 28] for retrieval and rank the paragraphs using BM25 with the same parameter settings ($k_1 = 0.9, b = 4$). The retriever returns 100 paragraphs and then feeds them to the BERT reader.

Statistics for the datasets are shown in Table 1. Note the possibly confusing terminology here: for SQuAD (as well as the other datasets), what we use for testing is actually the public development set (same as previous work). For reference, we also provide statistics for our distantly-supervised data; these represent the best settings, corresponding to our main results in Tables 2 and 3, which use a positive/negative ratio of 1:7, 1:3, 1:7, and 1:6, for each dataset, respectively, with random sampling (see Section 4.2).

For model training, we begin with the pre-trained BERT-Base model (uncased, 12-layer, 768-hidden, 12-heads, 110M parameters).

[1] https://pypi.org/project/hanziconv/0.2.1/

Table 1: Number of question–answer pairs in each dataset. DS(+) and DS(±) refer to our augmented dataset with positive and positive as well as negative examples, respectively, with the best setting from our main results.

	SQuAD	TriviaQA	CMRC	DRCD
Train	87,599	87,622	10,321	26,936
Test	10,570	11,313	3,351	3,524
DS(+)	64,244	264,192	8,596	41,792
DS(±)	447,468	789,089	68,696	246,604

We use the pre-trained Chinese BERT-Base for the Chinese datasets. All inputs are padded to 384 tokens; the learning rate is set to 3×10^{-5} and all other default settings are used. For all datasets, when applying stage-wise fine-tuning, we first fine-tune on the source dataset (SRC) for two epochs, and then on the augmented dataset for one epoch.

4 RESULTS

Main results on the two English datasets are shown in Table 2, along with comparisons directly copied from previous papers. Our figures are obtained from the dataset denoted DS(±) in Table 1; the training regime is the best configuration for combining different datasets, which will be discussed in the next section.[2] A few additional details: for SQuAD, we include results directly copied from the original BERTserini paper [29] (fine-tuning only on the source data, no data augmentation). Since then, however, we have made a few improvements, including engineering refinements and upgrading to a more recent version of Lucene. In the remainder of this paper, BERTserini refers to our improved implementation. When our distant supervision techniques were first proposed [30], we reported the best known score on SQuAD that we were aware of. Since then, however, the state of the art has further advanced, but those approaches take advantage of larger BERT models [1, 26], whereas we continued to use BERT-Base. Main results on the two Chinese datasets are shown in Table 3. To the best of our knowledge, there is no previous work on these two datasets in the retrieval-based setting, and therefore BERTserini is the only baseline available.

We can see that our distant supervision techniques and training regime for exploiting the augmented datasets lead to large improvements for all datasets, both in terms of exact match as well as F_1. Improvements in both English and Chinese suggest the generality of our approach. The statistical significance of all improvements (BERTserini + DS over BERTserini, across all four datasets) was verified with a paired t-test ($p << 0.01$). Note that we are unable to apply significance tests to the other conditions because we do not have access to results from those systems.

In Figure **??**, we show the effects of varying the number of paragraphs k that is fed into the BERT reader for SQuAD (top) and CMRC (bottom). We plot the following metrics:

(1) Top 1 Exact Match (EM): the top k paragraphs from the retriever are fed to the reader, the reader extracts the best phrase in each

[2] In later sections we present a number of detailed analyses and contrastive conditions under slightly different experimental procedures, and so there is no exact one-to-one correspondence between figures in the main results tables and subsequent results.

Table 2: Main results on the two English datasets.

SQuAD			
Model	EM	F_1	R
Dr.QA [4]	29.8	-	-
R^3 [24]	29.1	37.5	-
Kratzwald and Feuerriegel [11]	29.8	-	-
Par. R. [12]	30.2	-	-
Minimal [15]	34.7	42.5	64.0
ORQA [13]	34.7	-	64.0
RankQA [10]	35.8	-	-
DenSPI-Hybrid [20]	36.2	44.4	-
MUPPET [7]	39.3	46.2	-
BERTserini (2019) [29]	38.6	46.1	85.9
RE^3QA-Large [8]	41.9	50.2	-
Multi-passage BERT-Base [26]	51.2	59.0	-
Multi-passage BERT-Large [26]	53.0	60.9	-
GraphQA BERT-Large (wwm) [1]	56.5	63.8	-
BERTserini	41.8	49.5	86.3
BERTserini + DS	51.2	59.4	86.3
TriviaQA			
Model	EM	F_1	R
ORQA [13]	47.2	-	-
R^3 [24]	47.3	53.7	-
DS-QA [14]	48.7	56.3	-
Evidence Agg. [25]	50.6	57.3	-
BERTserini	51.0	56.3	83.7
BERTserini + DS	54.4	60.2	83.7

Table 3: Main results on the two Chinese datasets.

CMRC			
Model	EM	F_1	R
BERTserini	44.5	60.9	86.5
BERTserini + DS	48.6	64.6	86.5
DRCD			
Model	EM	F_1	R
BERTserini	50.7	65.0	81.5
BERTserini + DS	55.4	67.7	81.5

paragraph, and then the aggregator selects the best answer among all the candidates. If the best answer matches the ground truth, the system receives credit.

(2) Top k Exact Match (EM): the top k paragraphs from the retriever are fed to the reader, which extracts the best phrase in each paragraph. If *any* of these phrases matches the ground truth, the system receives credit.

(3) Recall: the fraction of questions in which at least one retrieved passage contains the answer (which provides an upper bound on scores with the current passage retrieval approach).

Looking at Figure **??**: as expected, all scores increase as more candidate paragraphs are fed to the reader, similar to what Yang et al. [29] observed. What is interesting, though, is to compare the effects of data augmentation, which we show with dotted vs. solid lines; note that the passage retriever (hence, recall) remains the same. For SQuAD (top), we see that top 1 EM improves quite a bit, although top k EM changes little. This suggests that distantly-supervised data is helping the model extract better answer phrases—likely due to improved context modeling—but isn't fundamentally expanding the coverage of the reader. We note the same phenomenon in Chinese (for CMRC, bottom). However, the gap here between top k EM and upper bound recall is much larger, which means that the passage retriever is finding relevant paragraphs, but the reader is unable to identify the correct answer spans. We suspect that the Chinese BERT model is weaker than the English BERT model,[3] and that the Chinese datasets lack diversity in both questions and answers compared to those in English.

Table 4: Results of exploring different approaches to combining source and augmented training data on the four datasets.

Model	EM	F_1	EM	F_1
	SQuAD		**TriviaQA**	
SRC	41.8	49.5	51.0	56.3
DS(+)	44.0	51.4	48.2	53.6
DS(±)	48.7	56.5	**54.4**	**60.2**
SRC+DS(±)	45.7	53.5	53.1	58.6
DS(±) → SRC	47.4	55.0	49.8	55.9
SRC → DS(±)	**50.2**	**58.2**	53.7	59.3
	CMRC		**DRCD**	
SRC	44.5	60.9	50.7	65.0
DS(+)	45.5	61.1	50.5	64.3
DS(±)	48.3	63.9	53.2	66.0
SRC+DS(±)	49.0	64.6	**55.4**	**67.7**
DS(±) → SRC	45.6	61.9	53.4	67.1
SRC → DS(±)	**49.2**	**65.4**	54.4	67.0

4.1 Fine-Tuning Order

As discussed in Section 2.2, having acquired the augmented data with our distant supervision techniques, there are a number of options for fine-tuning the model using all available data. Experiments exploring these possibilities are presented in Table 4 for each dataset. The rows marked SRC refer to fine-tuning with the source data only, which is the same as the BERTserini baselines in Tables 2 and 3. While training with positive examples, denoted DS(+), improves effectiveness as expected, an even larger gain comes from leveraging positive *and* negative examples, denoted DS(±).[4]

The natural question is how to take advantage of *both* types of data (source and augmented). The most obvious approach is to simply lump both the source and augmented data together to create a single, larger training set. This is shown in Table 4 as SRC + DS(±), and for three out of the four datasets, it is not the best approach (we discuss the exception below). In fact, for both English datasets, the SRC + DS(±) condition performs worse than just using the augmented data alone, i.e., DS(±) beats SRC + DS(±).

[3]The SRC-trained BERT-Base model achieves 80 EM and 88 F_1 on SQuAD 1.1, while only 65 EM and 83 F_1 on CMRC. These results are comparable to Sun et al. [22].

[4]Note that negative sampling here was performed using the top-down approach (see next section), which is slightly less effective than the random sampling approach presented in Tables 2 and 3; thus, these figures are not directly comparable.

We propose that heterogeneous datasets should be leveraged in a stage-wise manner, starting with the dataset that is "furthest" away from the test data. That is, we wish to take advantage of all available data, but the *last* dataset we use to fine-tune BERT should be "most like" the test data the model will encounter at evaluation time. Since the augmented data is drawn from the same passages that the system is ultimately evaluated on, we should fine-tune on those data *last*. This condition is denoted SRC → DS(±), which yields the best results for two of the datasets (SQuAD and CMRC); it is the second-best condition for the other two datasets (TriviaQA and DRCD). Note that if we swap the tuning order, DS(±) → SRC, effectiveness drops (and quite a bit in three of the four datasets), which lends credence to our heuristic.

For two of the datasets, our proposed stage-wise fine-tuning approach does not yield the highest effectiveness. With TriviaQA, the best condition is to simply disregard the source dataset, i.e., fine-tune with the augmented data only. We believe this may be an artifact of how the dataset was constructed to begin with: TriviaQA source paragraphs are already the product of distant supervision (on noisy web texts), which means that they are of lower quality to begin with. Our data augmentation procedure actually yields *higher quality* training data, since the examples are from Wikipedia and thus better match the passages that the BERT reader encounters at inference time.

For DRCD, we see that SRC + DS(±) is the most effective overall, which we explain by how the dataset is constructed. According to Shao et al. [21], the dataset was assembled by searching Wikipedia (10k paragraphs from 2.1k pages). In this case, the augmented data are similar enough to the source data that lumping them both together to create a single, larger dataset is the best strategy. This observation is affirmed by the fact that flipping the order of fine-tuning, DS(±) → SRC, yields only a small drop in effectiveness (much less than on the other datasets).

While the best conditions appear to be idiosyncratic for two of our datasets (artifacts of how they were constructed), we believe that our proposed stage-wise tuning approach makes intuitive sense. Another way to think about using different datasets is in terms of a very simple form of transfer learning. The stage-wise fine-tuning strategy is essentially trying to transfer knowledge from labeled data that is not drawn from the same distribution as the test instances. We wish to take advantage of transfer effects, but limit the scope of erroneous parameterization. Thus, it makes sense not to intermingle heterogeneous, qualitatively different datasets, but to fine-tune the model in distinct stages.

4.2 Negative Sampling Strategies

One key insight of our work is that negative sampling is critical to the effectiveness of distant supervision techniques, particularly in our retrieval-based setting where the reader is only exposed to the output of the passage retriever. In this section, we present results that explore different negative sampling strategies.

Table 5 shows the effectiveness of different negative sampling strategies (see Section 2.2) on two datasets: SQuAD for English and CMRC for Chinese. Due to space limitations, we limit our analyses to these two datasets only. In order to better isolate the impact of data augmentation, these experiments do not take advantage of the source data, and so the figures are not directly comparable to the main results in Tables 2 and 3. Recall that the top-down negative sampling strategy selects "difficult" examples (i.e., those with high BM25 scores); for English, this is more effective than bottom-up sampling, which selects "easy" examples. However, results suggest that random sampling beats both approaches, showing the importance of selecting diverse examples. Interestingly, though, for Chinese, the sampling strategy doesn't make much of a difference. This was a curious finding that prompted us to examine the retrieved paragraphs manually; as it turns out, passage retrieval in Chinese does not produce many paragraphs that are relevant to the topics discussed in the question, but do not contain the answer, because of our simple bigram-based approach. In other words, all of the negative examples are "easy", regardless of sampling strategy, and hence we observe little difference in effectiveness.

Table 5: Effects of different negative sampling strategies on SQuAD and CMRC.

	SQuAD		CMRC	
	EM	F_1	EM	F_1
Top-down	49.2	57.2	48.8	64.5
Bottom-up	46.8	54.9	48.6	65.2
Random	49.6	57.6	48.6	64.7

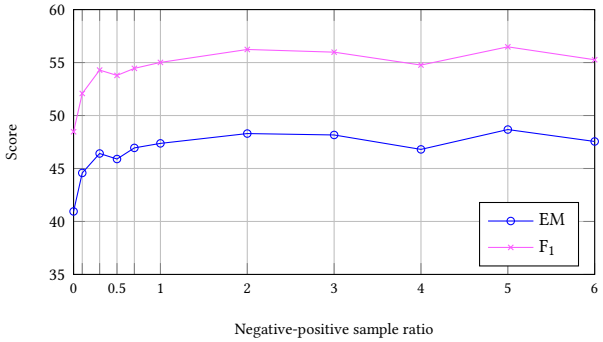

Figure 2: Effects of varying d, the positive–negative ratio of examples, on SQuAD.

As we discussed in Section 2.2, the ratio between positive and negative examples that are generated via our distant supervision technique is an important hyperparameter. In Figure 2, we plot the effect of this ratio d on EM and F_1. That is, for every positive example, we select d negative examples. We used a similar setup as the experiment above, and therefore these results are also not directly comparable to the main results in Tables 2 and 3. Furthermore, in order to experiment with $d < 1$, we applied a slightly different approach to negative sampling: Instead of pairing negative examples with positive examples (per Section 2.2), we first gathered all positive and negative examples across all questions, and then performed the sampling on those two groups given the ratio d. These results appear somewhat noisy, but suggest that $d > 2$ yields good effectiveness—and from the perspective of minimizing training time, we desire the smallest d value that allows us to fully

Table 6: Breakdown of effectiveness by question type for SQuAD (left) and CMRC (right).

	SQuAD					CMRC				
		BERTserini		BERTserini + DS			BERTserini		BERTserini + DS	
Type	percentage	EM	F_1	EM	F_1	percentage	EM	F_1	EM	F_1
Who	13.0%	43.1	48.6	52.9 (+22.7%)	59.3 (+22.0%)	8.0%	47.9	62.6	46.3 (−3.2%)	63.4 (+1.3%)
When	7.9%	46.8	54.7	56.9 (+21.6%)	65.0 (+18.8%)	6.0%	50.0	67.5	58.8 (+17.5%)	73.3 (+8.6%)
Where	4.7%	28.7	38.2	41.1 (+43.2%)	51.4 (+34.6%)	10.5%	43.6	61.3	52.2 (+19.7%)	67.1 (+9.4%)
What	54.8%	42.9	50.6	51.8 (+20.7%)	60.3 (+19.2%)	48.0%	40.8	55.8	44.3 (+8.6%)	59.2 (+6.2%)
Which	6.3%	46.1	53.5	58.6 (+27.1%)	65.8 (+23.0%)	13.2%	57.6	71.3	63.8 (+10.6%)	76.3 (+7.0%)
How	11.0%	36.6	44.5	45.0 (+23.0%)	52.5 (+18.0%)	8.4%	43.4	63.8	48.5 (+11.9%)	69.8 (+9.5%)
Why	1.4%	34.2	51.4	36.2 (+5.8%)	54.2 (+5.4%)	5.3%	34.3	55.5	32.6 (−5.1%)	53.9 (−2.8%)
Other	0.9%	28.9	49.5	40.0 (+38.4%)	52.6 (+6.3%)	0.4%	35.7	53.0	35.7 (+0.0%)	45.3 (−14.5%)

exploit distant supervision. The same experiments on the other three datasets yield similar conclusions.

4.3 Detailed Analyses

We conclude with detailed analyses. In Table 6, we break down effectiveness by question type for SQuAD (left) and CMRC (right), comparing the BERTserini baseline and the model trained with data augmentation (corresponding to the main results tables). Question classification in English is performed based on simple question word matching. For consistency, we translated all Chinese questions into English with a state-of-the-art machine translation system, and applied the same classification scheme as the English questions. We manually examined the translations of the Chinese questions and confirmed that they look reasonable.

For SQuAD, we see that effectiveness on "how" and "why" questions are lower than the other categories (which makes sense since they are more difficult), but surprisingly the scores for "where" questions are quite low as well. We see that data augmentation improves effectiveness across the board, although gains from "why" questions are quite limited—this is likely because the exact phrasing of the ground truth answers for these questions are difficult to find. For Chinese, "why" questions appear more difficult than the other types (not surprising). Interestingly, we see that data augmentation actually decreases effectiveness for some question types; however, most categories do see large gains.

How is data augmentation improving the reader? At a high-level, we believe that greater diversity in answers gives the reader a more accurate model of answer contexts. For example, we have noticed an artifact of how some questions are constructed in SQuAD: often, a question that contains "what *noun*" is directly answered by "X *noun*" in a text span. As a result, a model trained solely on SQuAD learns (basically) to pattern match, and frequently marks the token before the noun (i.e., X) as the answer, even in obviously incorrect contexts. With distant supervision, this is avoided because of exposure to sufficient answer diversity.

5 CONCLUSIONS

This paper explores the design space of distant supervision techniques for retrieval-based question answering from Wikipedia across two English and two Chinese datasets. Our two key insights are to exploit heterogeneous data in stage-wise fine-tuning and proper settings for negative sampling. A series of thorough experiments explore the impact of these various aspects.

A noteworthy advantage of our approach is its simplicity and the fact that our gains are demonstrated on a simple reader model. This means that, essentially, gains come "for free". Furthermore, our techniques should be applicable to other neural models as well, including improvements to the basic BERTserini model architecture proposed by other researchers [1, 7, 8, 13, 20, 26]. These findings confirm perhaps something that machine learning practitioners already know too well: "there's no data like more data".

REFERENCES

[1] Akari Asai, Kazuma Hashimoto, Hannaneh Hajishirzi, Richard Socher, and Caiming Xiong. 2019. Learning to Retrieve Reasoning Paths over Wikipedia Graph for Question Answering. *arXiv:1911.10470* (2019).

[2] Payal Bajaj, Daniel Campos, Nick Craswell, Li Deng, Jianfeng Gao, Xiaodong Liu, Rangan Majumder, Andrew McNamara, Bhaskar Mitra, Tri Nguyen, Mir Rosenberg, Xia Song, Alina Stoica, Saurabh Tiwary, and Tong Wang. 2016. MS MARCO: A Human Generated MAchine Reading COmprehension Dataset. *arXiv:1611.09268* (2016).

[3] Antoine Bordes, Nicolas Usunier, Sumit Chopra, and Jason Weston. 2015. Large-scale Simple Question Answering with Memory Networks. *arXiv:1506.02075* (2015).

[4] 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)*. Vancouver, Canada, 1870–1879.

[5] Yiming Cui, Ting Liu, Li Xiao, Zhipeng Chen, Wentao Ma, Wanxiang Che, Shijin Wang, and Guoping Hu. 2018. A Span-Extraction Dataset for Chinese Machine Reading Comprehension. *arXiv:1810.07366* (2018).

[6] 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)*. Minneapolis, Minnesota, 4171–4186.

[7] Yair Feldman and Ran El-Yaniv. 2019. Multi-Hop Paragraph Retrieval for Open-Domain Question Answering. *arXiv:1906.06606* (2019).

[8] Minghao Hu, Yuxing Peng, Zhen Huang, and Dongsheng Li. 2019. Retrieve, Read, Rerank: Towards End-to-End Multi-Document Reading Comprehension. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Florence, Italy, 2285–2295.

[9] 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)*. Vancouver, Canada, 1601–1611.

[10] Bernhard Kratzwald, Anna Eigenmann, and Stefan Feuerriegel. 2019. RankQA: Neural Question Answering with Answer Re-Ranking. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Florence, Italy, 6076–6085.

[11] Bernhard Kratzwald and Stefan Feuerriegel. 2018. Adaptive Document Retrieval for Deep Question Answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Brussels, Belgium, 576–581.

[12] Jinhyuk Lee, Seongjun Yun, Hyunjae Kim, Miyoung Ko, and Jaewoo Kang. 2018. Ranking Paragraphs for Improving Answer Recall in Open-Domain Question Answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Brussels, Belgium, 565–569.

[13] 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*. Florence, Italy, 6086–6096.

[14] Yankai Lin, Haozhe Ji, Zhiyuan Liu, and Maosong Sun. 2018. Denoising Distantly Supervised Open-Domain Question Answering. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Melbourne, Australia, 1736–1745.

[15] Sewon Min, Victor Zhong, Richard Socher, and Caiming Xiong. 2018. Efficient and Robust Question Answering from Minimal Context over Documents. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Melbourne, Australia, 1725–1735.

[16] Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep Contextualized Word Representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*. New Orleans, Louisiana, 2227–2237.

[17] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. *Improving Language Understanding by Generative Pre-training*. Technical Report.

[18] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ Questions for Machine Comprehension of Text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. Austin, Texas, 2383–2392.

[19] Ellen Riloff. 1996. Automatically Generating Extraction Patterns from Untagged Text. In *Proceedings of the Thirteenth National Conference on Artificial Intelligence and Eighth Innovative Applications of Artificial Intelligence Conference, Volume 2*. Portland, Oregon, 1044–1049.

[20] 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*. Florence, Italy, 4430–4441.

[21] Chih Chieh Shao, Trois Liu, Yuting Lai, Yiying Tseng, and Sam Tsai. 2018. DRCD: A Chinese Machine Reading Comprehension Dataset. *arXiv:1806.00920* (2018).

[22] Yu Sun, Shuohuan Wang, Yukun Li, Shikun Feng, Hao Tian, Hua Wu, and Haifeng Wang. 2019. ERNIE 2.0: A Continual Pre-training Framework for Language Understanding. *arXiv:1907.12412* (2019).

[23] Ellen M. Voorhees and Dawn M. Tice. 1999. The TREC-8 Question Answering Track Evaluation. In *Proceedings of the Eighth Text REtrieval Conference (TREC-8)*. Gaithersburg, Maryland, 83–106.

[24] Shuohang Wang, Mo Yu, Xiaoxiao Guo, Zhiguo Wang, Tim Klinger, Wei Zhang, Shiyu Chang, Gerald Tesauro, Bowen Zhou, and Jing Jiang. 2017. R^3: Reinforced Reader-Ranker for Open-Domain Question Answering. *arXiv:1709.00023* (2017).

[25] Shuohang Wang, Mo Yu, Jing Jiang, Wei Zhang, Xiaoxiao Guo, Shiyu Chang, Zhiguo Wang, Tim Klinger, Gerald Tesauro, and Murray Campbell. 2018. Evidence Aggregation for Answer Re-Ranking in Open-Domain Question Answering. *arXiv:1711.05116* (2018).

[26] Zhiguo Wang, Patrick Ng, Xiaofei Ma, Ramesh Nallapati, and Bing Xiang. 2019. Multi-passage BERT: A Globally Normalized BERT Model for Open-domain Question Answering. *arXiv:1908.08167* (2019).

[27] Peilin Yang, Hui Fang, and Jimmy Lin. 2017. Anserini: Enabling the Use of Lucene for Information Retrieval Research. In *Proceedings of the 40th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2017)*. Tokyo, Japan, 1253–1256.

[28] Peilin Yang, Hui Fang, and Jimmy Lin. 2018. Anserini: Reproducible Ranking Baselines Using Lucene. *Journal of Data and Information Quality* 10, 4 (2018), Article 16.

[29] Wei Yang, Yuqing Xie, Aileen Lin, Xingyu Li, Luchen Tan, Kun Xiong, Ming Li, and Jimmy Lin. 2019. End-to-End Open-Domain Question Answering with BERTserini. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*. Minneapolis, Minnesota, 72–77.

[30] Wei Yang, Yuqing Xie, Luchen Tan, Kun Xiong, Ming Li, and Jimmy Lin. 2019. Data Augmentation for BERT Fine-Tuning in Open-Domain Question Answering. *arXiv:1904.06652* (2019).

[31] 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*. Lisbon, Portugal, 2013–2018.

[32] Xuchen Yao, Benjamin Van Durme, Chris Callison-Burch, and Peter Clark. 2013. Answer Extraction as Sequence Tagging with Tree Edit Distance. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Atlanta, Georgia, 858–867.

[33] David Yarowsky. 1995. Unsupervised Word Sense Disambiguation Rivaling Supervised Methods. In *Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics*. Cambridge, Massachusetts, 189–196.