Latent Retrieval for Weakly Supervised Open Domain Question Answering

Kenton Lee **Ming-Wei Chang** **Kristina Toutanova**
Google Research
Seattle, WA
{kentonl,mingweichang,kristout}@google.com

arXiv:1906.00300v3 [cs.CL] 27 Jun 2019

Abstract

Recent work on open domain question answering (QA) assumes strong supervision of the supporting evidence and/or assumes a blackbox information retrieval (IR) system to retrieve evidence candidates. We argue that both are suboptimal, since gold evidence is not always available, and QA is fundamentally different from IR. We show for the first time that it is possible to jointly learn the retriever and reader from question-answer string pairs and without any IR system. In this setting, evidence retrieval from all of Wikipedia is treated as a latent variable. Since this is impractical to learn from scratch, we pre-train the retriever with an *Inverse Cloze Task*. We evaluate on open versions of five QA datasets. On datasets where the questioner already knows the answer, a traditional IR system such as BM25 is sufficient. On datasets where a user is genuinely seeking an answer, we show that learned retrieval is crucial, outperforming BM25 by up to 19 points in exact match.

1 Introduction

Due to recent advances in reading comprehension systems, there has been a revival of interest in open domain question answering (QA), where the evidence must be retrieved from an open corpus, rather than being given as input. This presents a more realistic scenario for practical applications.

Current approaches require a blackbox information retrieval (IR) system to do much of the heavy lifting, even though it cannot be fine-tuned on the downstream task. In the strongly supervised setting popularized by DrQA (Chen et al., 2017), they also assume a reading comprehension model trained on question-answer-evidence triples, such as SQuAD (Rajpurkar et al., 2016). The IR system is used at test time to generate evidence candidates in place of the gold evidence. In the weakly supervised setting, proposed by TriviaQA (Joshi et al., 2017), SearchQA (Dunn et al., 2017), and Quasar (Dhingra et al., 2017), the dependency on strong supervision is removed by assuming that the IR system provides noisy gold evidence.

These approaches rely on the IR system to massively reduce the search space and/or reduce spurious ambiguity. However, QA is fundamentally different from IR (Singh, 2012). Whereas IR is concerned with lexical and semantic matching, questions are *by definition* under-specified and require more language understanding, since users are explicitly looking for unknown information. Instead of being subject to the recall ceiling from blackbox IR systems, we should directly learn to retrieve using question-answering data.

In this work, we introduce the first *Open-Retrieval Question Answering* system (ORQA). ORQA learns to retrieve evidence from an open corpus, and is supervised only by question-answer string pairs. While recent work on improving evidence retrieval has made significant progress (Wang et al., 2018; Kratzwald and Feuerriegel, 2018; Lee et al., 2018; Das et al., 2019), they still only rerank a closed evidence set. The main challenge to fully end-to-end learning is that retrieval over the open corpus must be considered a latent variable that would be impractical to train from scratch. IR systems offer a reasonable but potentially suboptimal starting point.

The key insight of this work is that end-to-end learning *is* possible if we pre-train the retriever with an unsupervised *Inverse Cloze Task* (ICT). In ICT, a sentence is treated as a pseudo-question, and its context is treated as pseudo-evidence. Given a pseudo-question, ICT requires selecting the corresponding pseudo-evidence out of the candidates in a batch. ICT pre-training provides a sufficiently strong initialization such that ORQA, a joint retriever and reader model, can be fine-tuned end-to-end by simply optimiz-

Task	Training		Evaluation		Example
	Evidence	Answer	Evidence	Answer	
Reading Comprehension	given	span	given	string	SQuAD (Rajpurkar et al., 2016)
Open-domain QA					
Unsupervised QA	none	none	none	string	GPT-2 (Radford et al., 2019)
Strongly Supervised QA	given	span	heuristic	string	DrQA (Chen et al., 2017)
Weakly Supervised QA					
Closed Retrieval QA	heuristic	string	heuristic	string	TriviaQA (Joshi et al., 2017)
Open Retrieval QA	**learned**	string	**learned**	string	**ORQA (this work)**

Table 1: Comparison of assumptions made by related tasks, along with references to examples. Heuristic evidence refers to the typical strategy of considering only a closed set of evidence documents from a traditional IR system, which sets a strict upper-bound on task performance. In this work (ORQA), only question-answer string pairs are observed during training, and evidence retrieval is learned in a completely end-to-end manner.

ing the marginal log-likelihood of correct answers that were found.

We evaluate ORQA on open versions of five existing QA datasets. On datasets where the question writers already know the answer—SQuAD (Rajpurkar et al., 2016) and TriviaQA (Joshi et al., 2017)—the retrieval problem resembles traditional IR, and BM25 (Robertson et al., 2009) provides state-of-the-art retrieval. On datasets where question writers do not know the answer—Natural Questions (Kwiatkowski et al., 2019), WebQuestions (Berant et al., 2013), and CuratedTrec (Baudis and Sedivý, 2015)—we show that learned retrieval is crucial, providing improvements of 6 to 19 points in exact match over BM25.

2 Overview

In this section, we introduce notation for open domain QA that is useful for comparing prior work, baselines, and our proposed model.

2.1 Task

In open domain question answering, the input q is a question string, and the output a is an answer string. Unlike reading comprehension, the source of evidence is a modeling choice rather than a part of the task definition. We compare the assumptions made by variants of reading comprehension and question answering tasks in Table 1.

Evaluation is exact match with any of the reference answer strings after minor normalization such as lowercasing, following evaluation scripts from DrQA (Chen et al., 2017).

2.2 Formal Definitions

We introduce several general definitions of model components that subsume many retrieval-based open domain question answering systems.

Models are defined with respect to an unstructured text corpus that is split into B blocks of evidence texts. An answer derivation is a pair (b, s), where $1 \leq b \leq B$ indicates the index of an evidence block and s denotes a span of text within block b. The start and end token indices of span s are denoted by START(s) and END(s) respectively.

Models define a scoring function $S(b, s, q)$ indicating the goodness of an answer derivation (b, s) given a question q. Typically, this scoring function is decomposed over a retrieval component $S_{retr}(b, q)$ and a reader component $S_{read}(b, s, q)$:

$$S(b, s, q) = S_{retr}(b, q) + S_{read}(b, s, q)$$

During inference, the model outputs the answer string of the highest scoring derivation:

$$a^* = \text{TEXT}(\underset{b,s}{\text{argmax}}\, S(b, s, q))$$

where $\text{TEXT}(b, s)$ deterministically maps answer derivation (b, s) to an answer string. A major challenge of any open domain question answering system is handling the scale. In our experiments on the English Wikipedia corpus, we consider over 13 million evidence blocks b, each with over 2000 possible answer spans s.

2.3 Existing Pipelined Models

In existing retrieval-based open domain question answering systems, a blackbox IR system first chooses a closed set of evidence candidates. For example, the score from the retriever component of DrQA (Chen et al., 2017) is defined as:

$$S_{retr}(b, q) = \begin{cases} 0 & b \in \text{TOP}(k, \text{TF-IDF}(q, b)) \\ -\infty & \text{otherwise} \end{cases}$$

Most work following DrQA use the same candidates from TF-IDF and focus on reading comprehension or re-ranking. The reading component

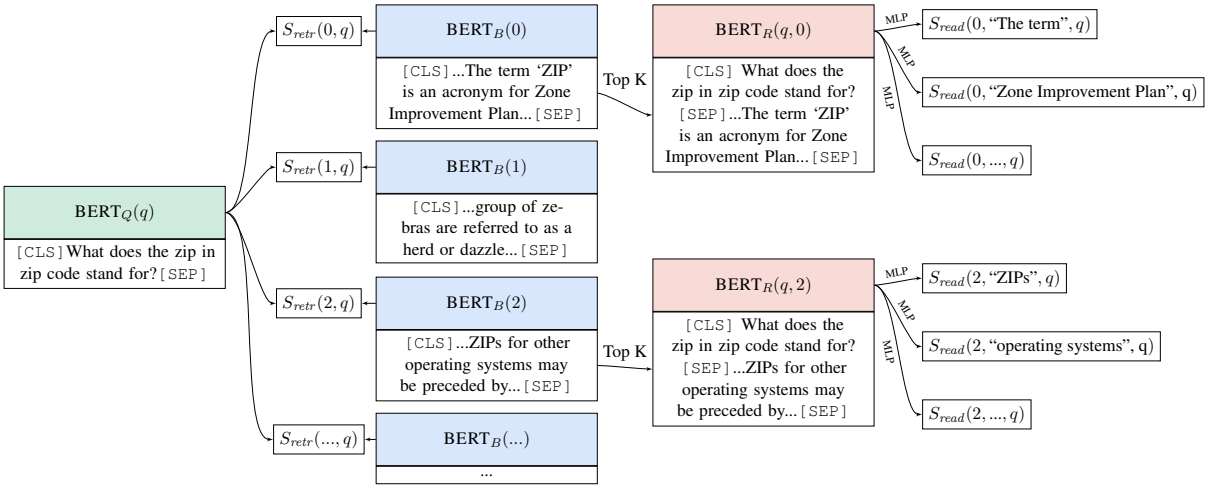

Figure 1: Overview of ORQA. A subset of all possible answer derivations given a question q is shown here. Retrieval scores $S_{retr}(q,b)$ are computed via inner products between BERT-based encoders. Top-scoring evidence blocks are jointly encoded with the question, and span representations are scored with a multi-layer perceptron (MLP) to compute $S_{read}(q,b,s)$. The final joint model score is $S_{retr}(q,b)$ + $S_{read}(q,b,s)$. Unlike previous work using IR systems for candidate proposal, we learn to retrieve from all of Wikipedia directly.

$S_{read}(b,s,q)$ is learned from gold answer derivations, typically from the SQuAD (Rajpurkar et al., 2016) dataset, where the evidence text is given.

In work that is more closely related to our approach, the reader is learned entirely from weak supervision (Joshi et al., 2017; Dhingra et al., 2017; Dunn et al., 2017). Spurious ambiguities (see Table 2) are heuristically removed by the retrieval system, and the cleaned results are treated as gold derivations.

3 Open-Retrieval Question Answering (ORQA)

We propose an end-to-end model where the retriever and reader components are jointly learned, which we refer to as the Open-Retrieval Question Answering (ORQA) model. An important aspect of ORQA is its expressivity—it is capable of retrieving any text in an open corpus, rather than being limited to the closed set returned by a black-box IR system. An illustration of how ORQA scores answer derivations is presented in Figure 1.

Following recent advances in transfer learning, all scoring components are derived from BERT (Devlin et al., 2018), a bidirectional transformer that has been pre-trained on unsupervised language-modeling data. We refer the reader to the original paper for details of the architecture. In this work, the relevant abstraction can be described by the following function:

$$\text{BERT}(x_1, [x_2]) = \{\texttt{CLS}: h_{\texttt{CLS}}, 1: h_1, 2: h_2, ...\}$$

The BERT function takes one or two string inputs (x_1 and optionally x_2) as arguments. It returns vectors corresponding to representations of the CLS pooling token or the input tokens.

Retriever component In order for the retriever to be learnable, we define the retrieval score as the inner product of dense vector representations of the question q and the evidence block b.

$$\begin{aligned} h_q &= \mathbf{W_q}\text{BERT}_Q(q)[\texttt{CLS}] \\ h_b &= \mathbf{W_b}\text{BERT}_B(b)[\texttt{CLS}] \\ S_{retr}(b,q) &= h_q^\top h_b \end{aligned}$$

where $\mathbf{W_q}$ and $\mathbf{W_b}$ are matrices that project the BERT output into 128-dimensional vectors.

Reader component The reader is a span-based variant of the reading comprehension model proposed in Devlin et al. (2018):

$$\begin{aligned} h_{start} &= \text{BERT}_R(q,b)[\text{START}(s)] \\ h_{end} &= \text{BERT}_R(q,b)[\text{END}(s)] \\ S_{read}(b,s,q) &= \text{MLP}([h_{start}; h_{end}]) \end{aligned}$$

Following Lee et al. (2016), a span is represented by the concatenation of its end points, which is scored by a multi-layer perceptron to enable start/end interaction.

Inference & Learning Challenges The model described above is conceptually simple. However, inference and learning are challenging since (1) an

Example	Supportive Evidence	Spurious Ambiguity
Q: Who is credited with developing the XY coordinate plane? **A**: René Descartes	...invention of Cartesian coordinates by **René Descartes** revolutionized...	...**René Descartes** was born in La Haye en Touraine, France...
Q: How many districts are in the state of Alabama? **A**: seven	...Alabama is currently divided into **seven** congressional districts, each represented by ...	...Alabama is one of **seven** states that levy a tax on food at the same rate as other goods...

Table 2: Examples of spurious ambiguities arising from the use of weak supervision. Good evidence retrieval is needed to generate a meaningful learning signal.

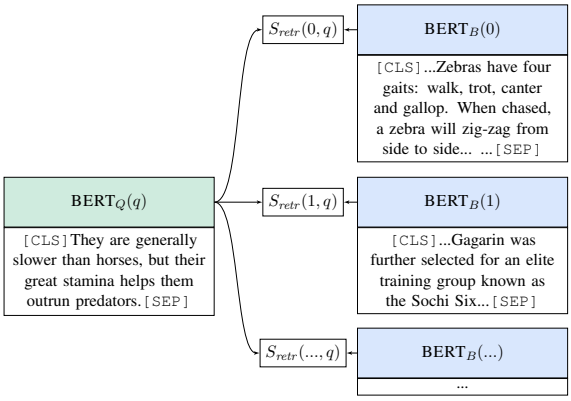

Figure 2: Example of the Inverse Cloze Task (ICT), used for retrieval pre-training. A random sentence (pseudo-query) and its context (pseudo evidence text) are derived from the text snippet: *"...Zebras have four gaits: walk, trot, canter and gallop.* ***They are generally slower than horses, but their great stamina helps them outrun predators****. When chased, a zebra will zig-zag from side to side..."* The objective is to select the true context among candidates in the batch.

open evidence corpus presents an enormous search space (over 13 million evidence blocks), and (2) how to navigate this space is entirely latent, so standard teacher-forcing approaches do not apply. Latent-variable methods are also difficult to apply naively due to the large number of spuriously ambiguous derivations. For example, as shown in Table 2, many irrelevant passages in Wikipedia would contain the answer string "seven."

We address these challenges by carefully initializing the retriever with unsupervised pre-training (Section 4). The pre-trained retriever allows us to (1) pre-encode all evidence blocks from Wikipedia, enabling dynamic yet fast top-k retrieval during fine-tuning (Section 5), and (2) bias the retrieval away from spurious ambiguities and towards supportive evidence (Section 6).

4 Inverse Cloze Task

The goal of our proposed pre-training procedure is for the retriever to solve an unsupervised task that closely resembles evidence retrieval for QA.

Intuitively, useful evidence typically discusses entities, events, and relations from the question. It also contains extra information (the answer) that is *not* present in the question. An unsupervised analog of a question-evidence pair is a sentence-context pair—the context of a sentence is semantically relevant and can be used to infer information missing from the sentence.

Following this intuition, we propose to pre-train our retrieval module with an Inverse Cloze Task (ICT). In the standard Cloze task (Taylor, 1953), the goal is to predict masked-out text based on its context. ICT instead requires predicting the inverse—given a sentence, predict its context (see Figure 2). We use a discriminative objective that is analogous to downstream retrieval:

$$P_{\text{ICT}}(b|q) = \frac{\exp(S_{retr}(b, q))}{\sum_{b' \in \text{BATCH}} \exp(S_{retr}(b', q))}$$

where q is a random sentence that is treated as a pseudo-question, b is the text surrounding q, and BATCH is the set of evidence blocks in the batch that are used as sampled negatives.

An important aspect of ICT is that it requires learning more than word matching features, since the pseudo-question is not present in the evidence. For example, the pseudo-question in Figure 2 never explicitly mentions "Zebras", but the retriever must still be able to select the context that discusses Zebras. Being able to infer the semantics from under-specified language is what sets QA apart from traditional IR.

However, we also do not want to dissuade the retriever from learning to perform word matching—lexical overlap is ultimately a very useful feature for retrieval. Therefore, we only remove the sentence from its context in 90% of the examples, encouraging the model to learn both abstract representations when needed and low-level word matching features when available.

ICT pre-training accomplishes two main goals:

1. Despite the mismatch between sentences dur-

ing pre-training and questions during fine-tuning, we expect *zero-shot* evidence retrieval performance to be sufficient for bootstrapping the latent-variable learning.

2. There is no such mismatch between pre-trained evidence blocks and downstream evidence blocks. We can expect the block encoder $\text{BERT}_B(b)$ to work well without further training. Only the question encoder needs to be fine-tuned on downstream data.

As we will see in the following section, these two properties are crucial for enabling computationally feasible inference and end-to-end learning.

5 Inference

Since fixed block encoders already provide a useful representation for retrieval, we can pre-compute all block encodings in the evidence corpus. As a result, the enormous set of evidence blocks does not need to be re-encoded while fine-tuning, and it can be pre-compiled into an index for fast maximum inner product search using existing tools such as Locality Sensitive Hashing.

With the pre-compiled index, inference follows a standard beam-search procedure. We retrieve the top-k evidence blocks and only compute the expensive reader scores for those k blocks. While we only consider the top-k evidence blocks during a single inference step, this set dynamically changes during training since the question encoder is fine-tuned according to the weakly supervised QA data, as discussed in the following section.

6 Learning

Learning is relatively straightforward, since ICT should provide non-trivial zero-shot retrieval. We first define a distribution over answer derivations:

$$P(b, s|q) = \frac{\exp(S(b, s, q))}{\sum_{b' \in \text{TOP}(k)} \sum_{s' \in b'} \exp(S(b', s', q))}$$

where $\text{TOP}(k)$ denotes the top k retrieved blocks based on S_{retr}. We use $k = 5$ in our experiments.

Given a gold answer string a, we find all (possibly spuriously) correct derivations in the beam, and optimize their marginal log-likelihood:

$$L_{\text{full}}(q, a) = -\log \sum_{b \in \text{TOP}(k)} \sum_{s \in b,\ a = \text{TEXT}(s)} P'(b, s|q)$$

where $a = \text{TEXT}(s)$ indicates whether the answer string a matches exactly the span s.

To encourage more aggressive learning, we also include an early update, where we consider a larger set of c evidence blocks but only update the retrieval score, which is cheap to compute:

$$P_{\text{early}}(b|q) = \frac{\exp(S_{retr}(b, q))}{\sum_{b' \in \text{TOP}(c)} \exp(S_{retr}(b', q))}$$

$$L_{\text{early}}(q, a) = -\log \sum_{b \in \text{TOP}(c),\ a \in \text{TEXT}(b)} P_{\text{early}}(b|q)$$

where $a \in \text{TEXT}(b)$ indicates whether answer string a appears in evidence block b. We use $c = 5000$ in our experiments.

The final loss includes both updates:

$$L(q, a) = L_{\text{early}}(q, a) + L_{\text{full}}(q, a)$$

If no matching answers are found at all, then the example is discarded. While we would expect almost all examples to be discarded with random initialization, we discard less than 10% of examples in practice due to ICT pre-training.

As previously mentioned, we fine-tune all parameters except those in the evidence block encoder. Since the query encoder is trainable, the model can potentially learn to retrieve any evidence block. This expressivity is a crucial difference from blackbox IR systems, where recall can only be improved by retrieving more evidence.

7 Experimental Setup

7.1 Open Domain QA Datasets

We train and evaluate on data from 5 existing question answering or reading comprehension datasets. Not all of them are intended as open domain QA datasets in their original form, so we convert them to open formats, following DrQA (Chen et al., 2017). Each example in the open version of the datasets consists of a single question string and a set of reference answer strings.

Natural Questions contains question from aggregated queries to Google Search (Kwiatkowski et al., 2019). To gather an open version of this dataset, we only keep questions with short answers and discard the given evidence document. Answers with many tokens often resemble extractive snippets rather than canonical answers, so we discard answers with more than 5 tokens.

Dataset	Train	Dev	Test	Example Question	Example Answer
Natural Questions	79168	8757	3610	What does the zip in zip code stand for?	Zone Improvement Plan
WebQuestions	3417	361	2032	What airport is closer to downtown Houston?	William P. Hobby Airport
CuratedTrec	1353	133	694	What metal has the highest melting point?	Tungsten
TriviaQA	78785	8837	11313	What did L. Fran Baum, author of The Wonderful Wizard of Oz, call his home in Hollywood?	Ozcot
SQuAD	78713	8886	10570	Other than the Automobile Club of Southern California, what other AAA Auto Club chose to simplify the divide?	California State Automobile Association

Table 3: Statistics and examples for the datasets that we evaluate on. There are slightly differences from the original datasets as described in Section 7.1, since not all of them were intended to be used in the open setting.

WebQuestions contains questions that were sampled from the Google Suggest API (Berant et al., 2013). The answers are annotated with respect to Freebase, but we only keep the string representation of the entities.

CuratedTrec is a corpus of question-answer pairs derived from TREC QA data curated by Baudis and Sedivý (2015). The questions come from various sources of real queries, such as MSNSearch or AskJeeves logs, where the question askers do not observe any evidence documents (Voorhees, 2001).

TriviaQA is a collection of trivia question-answer pairs that were scraped from the web (Joshi et al., 2017). We use their unfiltered set and discard their distantly supervised evidence.

SQuAD was designed to be a reading comprehension dataset rather than an open domain QA dataset (Rajpurkar et al., 2016). Answer spans were selected from a Wikipedia paragraph, and the questions were written by annotators who were instructed to ask questions that are answered by a given answer in a given context.

On datasets where a development set does not exist, we randomly hold out 10% of the training data for development. On datasets where the test set is hidden, we also randomly hold out 10% of the training data for development, and use the original development set for testing (following DrQA). A summary of dataset statistics and examples are shown in Table 3.

7.2 Dataset Biases

Evaluating on this diverse set of question-answer pairs is crucial, because all existing datasets have inherent biases that are problematic for open domain QA systems with learned retrieval. These biases are summarized in Table 4.

In the Natural Questions, WebQuestions, and CuratedTrec, the question askers do not already know the answer. This accurately reflects a distribution of genuine information-seeking questions. However, annotators must separately find correct answers, which requires assistance from automatic tools and can introduce a moderate bias towards results from the tool.

In TriviaQA and SQuAD, automatic tools are not needed since the questions are written with known answers in mind. However, this introduces another set of biases that are arguably more problematic. Question writing is not motivated by an information need. This often results in many hints in the question that would not be present in naturally occurring questions, as shown in the examples in Table 3. This is particularly problematic for SQuAD, where the question askers are also prompted with a specific piece of evidence for the answer, leading to artificially large lexical overlap between the question and evidence.

Note that these are simply properties of the datasets rather than actionable criticisms—such data collection methods are necessary to scale up, and it is unclear how one could collect a truly unbiased dataset without impractical costs.

7.3 Implementation Details

We mainly evaluate in the setting where only question-answer string pairs are available for supervision. See Section 9 for head-to-head comparisons with the DrQA setting that uses the same evidence corpus and the same type of supervision.

Evidence Corpus We use the English Wikipedia snapshot from December 20, 2018 as the evidence corpus.[1] The corpus is greedily

[1] We deviate from DrQA's 2016 Wikipedia evidence corpus because the original snapshot is no longer publicly available. The 12-20-2018 snapshot is available at https://archive.org/download/enwiki-20181220.

Dataset	Question writer knows answer	Question writer knows evidence	Tool-assisted answer
Natural Questions			✓
WebQuestions			✓
CuratedTrec			✓
TriviaQA	✓		
SQuAD	✓	✓	

Table 4: A breakdown of biases in existing QA datasets. These biases are associated with either the question or the answer.

split into chunks of at most 288 wordpieces based on BERT's tokenizer, while preserving sentence boundaries. This results in just over 13 million evidence blocks. The title of the document is included in the block encoder.

Hyperparameters In all uses of BERT (both the retriever and reader), we initialize from the uncased base model, which consists of 12 transformer layers with a hidden size of 768.

As mentioned in Section 3, the retrieval representations, h_q and h_b, have 128 dimensions. The small hidden size was chosen so that the final QA model can comfortably run on a single machine. We use the default optimizer from BERT.

When pre-training the retriever with ICT, we use a learning rate of 10^{-4} and a batch size of 4096 on Google Cloud TPUs for 100k steps. When fine-tuning, we use a learning rate of 10^{-5} and a batch size of 1 on a single machine with a 12GB GPU. Answer spans are limited to 10 tokens. We perform 2 epochs of fine-tuning for the larger datasets (Natural Questions, TriviaQA, and SQuAD), and 20 epochs for the smaller datasets (WebQuestions and CuratedTrec).

8 Main Results

8.1 Baselines

We compare against other retrieval methods by using alternate retrieval scores $S_{retr}(b, q)$, but with the same reader.

BM25 A de-facto state-of-the-art unsupervised retrieval method is BM25 (Robertson et al., 2009). It has been shown to be robust for both traditional information retrieval tasks, and evidence retrieval for question answering (Yang et al., 2017).[2] Since

[2]We also include the title, which was slightly beneficial.

	Model	BM25 +BERT	NNLM +BERT	ELMo +BERT	ORQA
Dev	Natural Questions	24.8	3.2	3.6	**31.3**
	WebQuestions	20.8	9.1	17.7	**38.5**
	CuratedTrec	27.1	6.0	8.3	**36.8**
	TriviaQA	**47.2**	7.3	6.0	45.1
	SQuAD	**28.1**	2.8	1.9	26.5
Test	Natural Questions	26.5	4.0	4.7	**33.3**
	WebQuestions	17.7	7.3	15.6	**36.4**
	CuratedTrec	21.3	4.5	6.8	**30.1**
	TriviaQA	**47.1**	7.1	5.7	45.0
	SQuAD	**33.2**	3.2	2.3	20.2

Table 5: **Main results**: End-to-end exact match for open-domain question answering from question-answer pairs only. Datasets where question askers know the answer behave differently from datasets where they do not.

BM25 is not trainable, the retrieved evidence considered during fine-tuning is static. Inspired by BERTserini (Yang et al., 2019), the final score is a learned weighted sum of the BM25 and reader score. Our implementation is based on Lucene.[3]

Language Models While unsupervised neural retrieval is notoriously difficult to improve over traditional IR (Lin, 2019), we include them as baselines for comparison. We experiment with unsupervised pooled representations from neural language models (LM), which has been shown to be state-of-the-art unsupervised representations (Perone et al., 2018). We compare with two widely-used 128-dimensional representations: (1) NNLM, context-*independent* embeddings from a feed-forward LMs (Bengio et al., 2003),[4] and (2) ELMo (small), a context-*dependent* bidirectional LSTM (Peters et al., 2018).[5]

As with ICT, we use the alternate encoders to pre-compute the encoded evidence blocks h_b and to initialize the question encoding h_q, which is fine-tuned. Based on existing IR literature and the intuition that LMs do not explicitly optimize for retrieval, we do not expect these to be strong baselines, but they demonstrate the difficulty of encoding blocks of text into 128 dimensions.

8.2 Results

The main results are show in Table 5. The first result to note is that BM25 is a powerful retrieval system. Word matching is important, and

[3]https://lucene.apache.org/

[4]https://tfhub.dev/google/nnlm-en-dim128/1

[5]https://allennlp.org/elmo

Model	Evidence Retrieved	SQuAD
DRQA	5 documents	27.1
DRQA (DS)	5 documents	28.4
DRQA (DS + MTL)	5 documents	29.8
BERTSERINI	5 documents	19.1
BERTSERINI	29 paragraphs	36.6
BERTSERINI	100 paragraphs	38.6
BM25 + BERT (gold deriv.)	5 blocks	34.7

Table 6: **Analysis**: Results comparable to previous work in the strongly supervised setting, where models have access to gold derivations from SQuAD. Different systems segment Wikipedia differently. There are 5.1M documents, 29.5M paragraphs, and 12.1M blocks in the December 12, 2016 Wikipedia snapshot.

dense vector representations derived from language models do not readily capture this.

We also show that on questions that were derived from real users who are seeking information (Natural Questions, WebQuestions, and CuratedTrec), our ICT pre-trained retriever outperforms BM25 by a large marge—6 to 19 points in exact match depending on the dataset.

However, in datasets where the question askers already know the answer, i.e. SQuAD and TriviaQA, the retrieval problem resembles traditional IR. In this setting, a highly compressed 128-dimensional vector cannot match BM25's ability to precisely represent every word in the evidence.

The notable drop between development and test accuracy for SQuAD is a reflection of an artifact in the dataset—its 100k questions are derived from only 536 documents. Therefore, good retrieval targets are highly correlated between training examples, violating the IID assumption, and making it unsuitable for learned retrieval. We strongly suggest that those who are interested in end-to-end open-domain QA models no longer train and evaluate with SQuAD for this reason.

9 Analysis

9.1 Strongly supervised comparison

To verify that our BM25 baseline is indeed state of the art, we also provide direct comparisons with DrQA's setup, where systems have access to gold answer derivations from SQuAD (Rajpurkar et al., 2016). While many systems have been proposed following DrQA's original setting, we compare only to the original system and the best system that

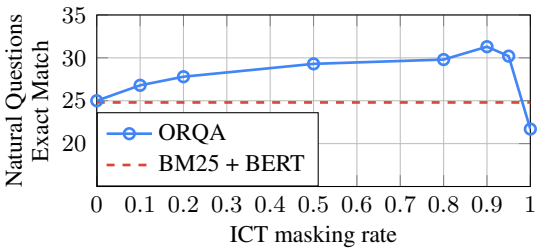

Figure 3: **Analysis**: Performance on our open version of the Natural Questions dev set with various masking rates for the ICT pre-training. Too much masking prevents the model from learning to exploit exact n-gram overlap. Too little masking makes language understanding unnecessary.

we are aware of—BERTserini (Yang et al., 2019).

DrQA's reader is DocReader (Chen et al., 2017), and they use TF-IDF to retrieve the top k documents. They also include distant supervision based on TF-IDF retrieval. BERTserini's reader is derived from base BERT (much like our reader), and they use BM25 to retrieve the top k paragraphs (much like our BM25 baseline). A major difference is that BERTserini uses true paragraphs from Wikipedia rather than arbitrary blocks, resulting in more evidence blocks due to uneven lengths.

For fair comparison with these strongly supervised systems, we pre-train the reader on SQuAD data.[6] In Table 6, our BM25 baseline, which retrieves 5 evidence blocks, greatly outperforms 5-document BERTserini and is close to 29-paragraph BERTserini.

9.2 Masking Rate in the Inverse Cloze Task

The pseudo-query is masked from the evidence block 90% of the time, motivated by intuition in Section 4. We empirically verify our intuitions in Figure 3 by varying the masking rate, and comparing results on our open version of the Natural Questions development set.

If we always mask the pseudo-query, the retriever never learns that n-gram overlap is a powerful retrieval signal, losing almost 10 points in end-to-end performance. If we never mask the pseudo-query, the problem is reduced to memorization and does not generalize well to question answering. The latter loses 6 points in end-to-end performance, which—perhaps not surprisingly—produces near-identical results to BM25.

[6]We use DrQA's December 12, 2016 snapshot of Wikipedia for an apples-to-apples comparison.

Example	ORQA	BM25 + BERT
Q: what is the new orleans saints symbol called **A**: fleur-de-lis	...The team's primary colors are old gold and black; their logo is a simplified **fleur-de-lis**. They played their home games in Tulane Stadium through the 1974 NFL season....	...the **SkyDome** was owned by Sportsco at the time... the sale of the New Orleans Saints with team owner Tom Benson... the Saints became a symbol for that community...
Q: how many senators per state in the us **A**: two	...powers of the Senate are established in Article One of the U.S. Constitution. Each U.S. state is represented by **two** senators...	...The Georgia Constitution mandates a maximum of **56** senators, elected from single-member districts...
Q: when was germany given a permanent seat on the council of the league of nations **A**: 1926	...Under the Weimar Republic, Germany (in fact the "Deutsches Reich" or German Empire) was admitted to the League of Nations through a resolution passed on September 8 **1926**. An additional 15 countries joined later...	...the accession of the German Democratic Republic to the Federal Republic of Germany, it was effective on **3 October 1990**...Germany has been elected as a non-permanent member of the United Nations Security Council...
Q: when was diary of a wimpy kid double down published **A**: November 1, 2016	..."Diary of a Wimpy Kid" first appeared on FunBrain in 2004, where it was read 20 million times. The abridged hardcover adaptation was released on **April 1, 2007**...	Diary of a Wimpy Kid: Double Down is the eleventh book in the "Diary of a Wimpy Kid" series by Jeff Kinney... The book was published on **November 1, 2016**...

Table 7: **Analysis**: Example predictions on our open version of the Natural Questions dev set. We show the highest scoring derivation, consisting of the evidence block and the predicted answer in bold. ORQA is more robust at separating semantically distinct text that have high lexical overlap. However, the limitation of the 128-dimensional vectors is that extremely specific concepts are less precisely represented.

9.3 Example Predictions

For a more intuitive understanding of the improvements from ORQA, we compare its predictions with baseline predictions in Table 7. We find that ORQA is more robust at separating semantically distinct text with high lexical overlap, as shown in the first three examples. However, it is expected that there are limits to how much information can be compressed into 128-dimensional vectors. The last example shows that ORQA has trouble precisely representing extremely specific concepts that sparse representations can cleanly separate. These errors indicate that a hybrid approach would be promising future work.

10 Related Work

Recent progress has been made towards improving evidence retrieval (Wang et al., 2018; Kratzwald and Feuerriegel, 2018; Lee et al., 2018; Das et al., 2019) by learning to aggregate from multiple retrieval steps. They re-rank evidence candidates from a closed set, and we aim to integrate these complementary approaches in future work.

Our approach is also reminiscent of weakly supervised semantic parsing (Clarke et al., 2010; Liang et al., 2013; Artzi and Zettlemoyer, 2013; Fader et al., 2014; Berant et al., 2013; Kwiatkowski et al., 2013), with which we share similar challenges—(1) inference and learning are tightly coupled, (2) latent derivations must be discovered, and (3) strong inductive biases are needed to find positive learning signal while avoiding spurious ambiguities.

While we motivate ICT from first principles as an unsupervised proxy for evidence retrieval, it is closely related to existing representation learning literature. ICT can be considered a generalization of the skip-gram objective (Mikolov et al., 2013), with a coarser granularity, deep architecture, and in-batch negative sampling from Logeswaran and Lee (2018).

Consulting external evidence sources with latent retrieval has also been explored in information extraction (Narasimhan et al., 2016). In comparison, we are able to learn a much more expressive retriever due to the strong inductive biases from ICT pre-training.

11 Conclusion

We presented ORQA, the first open domain question answering system where the retriever and reader are jointly learned end-to-end using only question-answer pairs and without any IR system. This is made possible by pre-training the retriever using an Inverse Cloze Task (ICT). Experiments show that learning to retrieve is crucial when the questions reflect an information need, i.e. the question writers do not already know the answer.

Acknowledgements

We thank the Google AI Language Team for valuable suggestions and feedback.

References

Yoav Artzi and Luke Zettlemoyer. 2013. Weakly supervised learning of semantic parsers for mapping instructions to actions. *Transactions of the Association for Computational Linguistics*, 1(1):49–62.

Petr Baudis and Jan Sedivý. 2015. Modeling of the question answering task in the yodaqa system. In *CLEF*.

Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Jauvin. 2003. A neural probabilistic language model. *Journal of machine learning research*, 3(Feb):1137–1155.

Jonathan Berant, Andrew Chou, Roy Frostig, and Percy Liang. 2013. Semantic parsing on freebase from question-answer pairs. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1533–1544.

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)*, volume 1, pages 1870–1879.

James Clarke, Dan Goldwasser, Ming-Wei Chang, and Dan Roth. 2010. Driving semantic parsing from the world's response. In *Proceedings of the fourteenth conference on computational natural language learning*, pages 18–27. Association for Computational Linguistics.

Rajarshi Das, Shehzaad Dhuliawala, Manzil Zaheer, and Andrew McCallum. 2019. Multi-step retriever-reader interaction for scalable open-domain question answering. In *International Conference on Learning Representations*.

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

Bhuwan Dhingra, Kathryn Mazaitis, and William W Cohen. 2017. Quasar: Datasets for question answering by search and reading. *arXiv preprint arXiv:1707.03904*.

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

Anthony Fader, Luke Zettlemoyer, and Oren Etzioni. 2014. Open question answering over curated and extracted knowledge bases. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1156–1165. ACM.

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)*, volume 1, pages 1601–1611.

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

Tom Kwiatkowski, Eunsol Choi, Yoav Artzi, and Luke Zettlemoyer. 2013. Scaling semantic parsers with on-the-fly ontology matching. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 1545–1556.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Rhinehart, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Matthew Kelcey, Jacob Devlin, et al. 2019. Natural questions: a benchmark for question answering research. *Transactions of the Association for Computational Linguistics*.

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

Kenton Lee, Shimi Salant, Tom Kwiatkowski, Ankur Parikh, Dipanjan Das, and Jonathan Berant. 2016. Learning recurrent span representations for extractive question answering. *arXiv preprint arXiv:1611.01436*.

Percy Liang, Michael I Jordan, and Dan Klein. 2013. Learning dependency-based compositional semantics. *Computational Linguistics*, 39(2):389–446.

Jimmy Lin. 2019. The neural hype and comparisons against weak baselines. In *ACM SIGIR Forum*.

Lajanugen Logeswaran and Honglak Lee. 2018. An efficient framework for learning sentence representations. *arXiv preprint arXiv:1803.02893*.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Karthik Narasimhan, Adam Yala, and Regina Barzilay. 2016. Improving information extraction by acquiring external evidence with reinforcement learning. *arXiv preprint arXiv:1603.07954*.

Christian S Perone, Roberto Silveira, and Thomas S Paula. 2018. Evaluation of sentence embeddings in downstream and linguistic probing tasks. *arXiv preprint arXiv:1806.06259*.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proc. of NAACL*.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI Blog*.

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*, pages 2383–2392.

Stephen Robertson, Hugo Zaragoza, et al. 2009. The probabilistic relevance framework: Bm25 and beyond. *Foundations and Trends in Information Retrieval*, 3(4):333–389.

Amit Singh. 2012. Entity based q&a retrieval. In *Proceedings of the 2012 Joint conference on empirical methods in natural language processing and computational natural language learning*, pages 1266–1277. Association for Computational Linguistics.

Wilson L Taylor. 1953. "Cloze procedure": A new tool for measuring readability. *Journalism Bulletin*, 30(4):415–433.

Ellen M Voorhees. 2001. Overview of the trec 2001 question answering track. In *In Proceedings of the Tenth Text REtrieval Conference (TREC*. Citeseer.

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

Peilin Yang, Hui Fang, and Jimmy Lin. 2017. Anserini: Enabling the use of lucene for information retrieval research. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1253–1256. ACM.

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. *arXiv preprint arXiv:1902.01718*.