Efficient and Robust Question Answering from Minimal Context over Documents

Sewon Min[1*] **Victor Zhong**[2]**, Richard Socher**[2]**, Caiming Xiong**[2]

Seoul National University[1], Salesforce Research[2]

shmsw25@snu.ac.kr,{vzhong, rsocher, cxiong}@salesforce.com

Abstract

Neural models for question answering (QA) over documents have achieved significant performance improvements. Although effective, these models do not scale to large corpora due to their complex modeling of interactions between the document and the question. Moreover, recent work has shown that such models are sensitive to adversarial inputs. In this paper, we study the minimal context required to answer the question, and find that most questions in existing datasets can be answered with a small set of sentences. Inspired by this observation, we propose a simple sentence selector to select the minimal set of sentences to feed into the QA model. Our overall system achieves significant reductions in training (up to 15 times) and inference times (up to 13 times), with accuracy comparable to or better than the state-of-the-art on SQuAD, NewsQA, TriviaQA and SQuAD-Open. Furthermore, our experimental results and analyses show that our approach is more robust to adversarial inputs.

1 Introduction

The task of textual question answering (QA), in which a machine reads a document and answers a question, is an important and challenging problem in natural language processing. Recent progress in performance of QA models has been largely due to the variety of available QA datasets (Richardson et al., 2013; Hermann et al., 2015; Rajpurkar et al., 2016; Trischler et al., 2016; Joshi et al., 2017; Kočiský et al., 2017).

Many neural QA models have been proposed for these datasets, the most successful of which tend to leverage coattention or bidirectional attention mechanisms that build codependent representations of the document and the question (Xiong et al., 2018; Seo et al., 2017).

Yet, learning the full context over the document is challenging and inefficient. In particular, when the model is given a long document, or multiple documents, learning the full context is intractably slow and hence difficult to scale to large corpora. In addition, Jia and Liang (2017) show that, given adversarial inputs, such models tend to focus on wrong parts of the context and produce incorrect answers.

In this paper, we aim to develop a QA system that is scalable to large documents as well as robust to adversarial inputs. First, we study the context required to answer the question by sampling examples in the dataset and carefully analyzing them. We find that most questions can be answered using a few sentences, without the consideration of context over entire document. In particular, we observe that on the SQuAD dataset (Rajpurkar et al., 2016), 92% of answerable questions can be answered using a single sentence.

Second, inspired by this observation, we propose a sentence selector to select the minimal set of sentences to give to the QA model in order to answer the question. Since the minimum number of sentences depends on the question, our sentence selector chooses a different number of sentences for each question, in contrast with previous models that select a fixed number of sentences. Our sentence selector leverages three simple techniques — weight transfer, data modification and score normalization, which we show to be highly effective on the task of sentence selection.

We compare the standard QA model given the *full* document (FULL) and the QA model given the

---

*All work was done while the author was an intern at Salesforce Research.

N sent	% on SQuAD	% on TriviaQA	Document	Question
1	90	56	**In 1873, Tesla returned to his birthtown, Smiljan.** Shortly after he arrived, (...)	Where did Tesla return to in 1873?
2	6	28	After leaving Edison's company Tesla partnered with two businessmen in 1886, Robert Lane and Benjamin Vail, who agreed to finance an electric lighting company in Tesla's name, Tesla Electric Light & Manufacturing. **The company installed electrical arc light based illumination systems designed by Tesla and also had designs for dynamo electric machine commutators, (...)**	What did Tesla Electric Light & Manufacturing do?
3↑	2	4	**Kenneth Swezey, a journalist whom Tesla had befriended, confirmed that Tesla rarely slept .** Swezey recalled one morning when Tesla called him at 3 a.m. : "I was sleeping in my room (...) Suddenly, the telephone ring awakened me ...	Who did Tesla call in the middle of the night?
N/A	2	12	**Writers whose papers are in the library are as diverse as Charles Dickens and Beatrix Potter.** Illuminated manuscripts in the library dating from (...)	The papers of which famous English Victorian author are collected in the library?

Table 1: Human analysis of the context required to answer questions on SQuAD and TriviaQA. 50 examples from each dataset are sampled randomly. 'N sent' indicates the number of sentences required to answer the question, and 'N/A' indicates the question is not answerable even given all sentences in the document. 'Document' and 'Question' are from the representative example from each category on SQuAD. Examples on TriviaQA are shown in Appendix B. The groundtruth answer span is in red text, and the oracle sentence (the sentence containing the grountruth answer span) is in **bold text**.

No.	Description	%	Sentence	Question
0	Correct (Not exactly same as grountruth)	58	**Gothic** architecture is represented in the majestic churches but also at the burgher houses and fortifications.	What type of architecture is represented in the majestic churches?
1	Fail to select precise span	6	Brownlee argues that disobedience in opposition to the decisions of non-governmental agencies such as **trade unions, banks, and private** universities can be justified if it reflects 'a larger challenge to the legal system that permits those decisions to be taken;.	Brownlee argues disobedience can be justified toward what institutions?
2	Complex semantics in sentence/question	34	Newton was limited by Denver's defense, which sacked him **seven** times and forced him into <u>three</u> turnovers, including a fumble which they recovered for a touchdown.	How many times did the Denver defense force Newton into turnovers?
3	Not answerable even with full paragraph	2	He encourages a distinction between lawful protest demonstration, **nonviolent** civil disobedience, and <u>violent</u> civil disobedience.	What type of civil disobedience is accompanied by aggression?

Table 2: Error cases (on exact match (EM)) of DCN+ given oracle sentence on SQuAD. 50 examples are sampled randomly. Grountruth span is in <u>underlined text</u>, and model's prediction is in **bold text**.

minimal set of sentences (MINIMAL) on five different QA tasks with varying sizes of documents. On SQuAD, NewsQA, TriviaQA(Wikipedia) and SQuAD-Open, MINIMAL achieves significant reductions in training and inference times (up to 15× and 13×, respectively), with accuracy comparable to or better than FULL. On three of those datasets, this improvements leads to the new state-of-the-art. In addition, our experimental results and analyses show that our approach is more robust to adversarial inputs. On the development set of SQuAD-Adversarial (Jia and Liang, 2017), MINIMAL outperforms the previous state-of-the-art model by up to 13%.

2 Task analyses

Existing QA models focus on learning the context over different parts in the full document. Although effective, learning the context within the full document is challenging and inefficient. Consequently, we study the minimal context in the document required to answer the question.

2.1 Human studies

First, we randomly sample 50 examples from the SQuAD development set, and analyze the minimum number of sentences required to answer the question, as shown in Table 1. We observed that 98% of questions are answerable given the document. The remaining 2% of questions are not answerable even given the entire document. For instance, in the last example in Table 1, the question requires the background knowledge that Charles Dickens is an English Victorian author. Among the answerable examples, 92% are answerable with a single sentence, 6% with two sentences, and 2% with three or more sentences.

We perform a similar analysis on the TriviaQA (Wikipedia) development (verified) set. Finding the sentences to answer the question on TriviaQA is more challenging than on SQuAD, since TriviaQA documents are much longer than SQuAD documents (488 vs 5 sentences per document). Nevertheless, we find that most examples are answerable with one or two sentences — among the 88% of examples that are answerable given the full document, 95% can be answered with one or two sentences.

2.2 Analyses on existing QA model

Given that the majority of examples are answerable with a single oracle sentence on SQuAD, we analyze the performance of an existing, competitive QA model when it is given the oracle sentence. We train DCN+ (Xiong et al., 2018), one of the state-of-the-art models on SQuAD (details in Section 3.1), on the oracle sentence. The model achieves 83.1 F1 when trained and evaluated using the full document and 85.1 F1 when trained and evaluated using the oracle sentence. We analyze 50 randomly sampled examples in which the model fails on exact match (EM) despite using the oracle sentence. We classify these errors into 4 categories, as shown in Table 2. In these examples, we observed that 40% of questions are answerable given the oracle sentence but the model unexpectedly fails to find the answer. 58% are those in which the model's prediction is correct but does not lexically match the groundtruth answer, as shown in the first example in Table 2. 2% are those in which the question is not answerable even given the full document. In addition, we compare predictions by the model trained using the full document (FULL) with the model trained on the oracle sentence (ORACLE). Figure 1 shows the Venn diagram of the questions answered correctly by FULL and ORACLE on SQuAD and NewsQA. ORACLE is able to answer 93% and 86% of the questions correctly answered by FULL on SQuAD and NewsQA, respectively.

These experiments and analyses indicate that if the model can accurately predict the oracle sentence, the model should be able to achieve comparable performance on overall QA task. Therefore, we aim to create an effective, efficient and robust QA system which only requires a single or a few sentences to answer the question.

3 Method

Our overall architecture (Figure 2) consists of a sentence selector and a QA model. The sentence selector computes a selection score for each sentence in parallel. We give to the QA model a reduced set of sentences with high selection scores to answer the question.

3.1 Neural Question Answering Model

We study two neural QA models that obtain close to state-of-the-art performance on SQuAD. **DCN+** (Xiong et al., 2018) is one of the start-
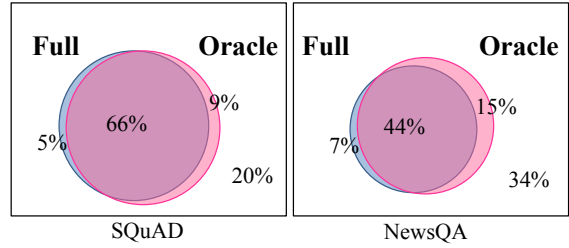

Figure 1: Venn diagram of the questions answered correctly (on exact match (EM)) by the model given a full document (FULL) and the model given an oracle sentence (ORACLE) on SQuAD (left) and NewsQA (right).

of-the-art QA models, achieving 83.1 F1 on the SQuAD development set. It features a deep residual coattention encoder, a dynamic pointing decoder, and a mixed objective that combines cross entropy loss with self-critical policy learning. **S-Reader** is another competitive QA model that is simpler and faster than DCN+, with 79.9 F1 on the SQuAD development set. It is a simplified version of the reader in DrQA (Chen et al., 2017), which obtains 78.8 F1 on the SQuAD development set. Model details and training procedures are shown in Appendix A.

3.2 Sentence Selector

Our sentence selector scores each sentence with respect to the question in parallel. The score indicates whether the question is answerable with this sentence.

The model architecture is divided into the encoder module and the decoder module. The encoder is a shared module with S-Reader, which computes sentence encodings and question encodings from the sentence and the question as inputs. First, the encoder computes sentence embeddings $D \in \mathbb{R}^{h_d \times L_d}$, question embeddings $Q \in \mathbb{R}^{h_d \times L_q}$, and question-aware sentence embeddings $D^q \in \mathbb{R}^{h_d \times L_d}$, where h_d is the dimension of word embeddings, and L_d and L_q are the sequence length of the document and the question, respectively. Specifically, question-aware sentence embeddings are obtained as follows.

$$\alpha_i = \text{softmax}(D_i^T W_1 Q) \in \mathbb{R}^{L_q} \quad (1)$$

$$D_i^q = \sum_{j=1}^{L_q} (\alpha_{i,j} Q_j) \in \mathbb{R}^{h_d} \quad (2)$$

Here, $D_i \in \mathbb{R}^{h_d}$ is the hidden state of sentence embedding for the i_{th} word and $W_1 \in \mathbb{R}^{h_d \times h_d}$ is
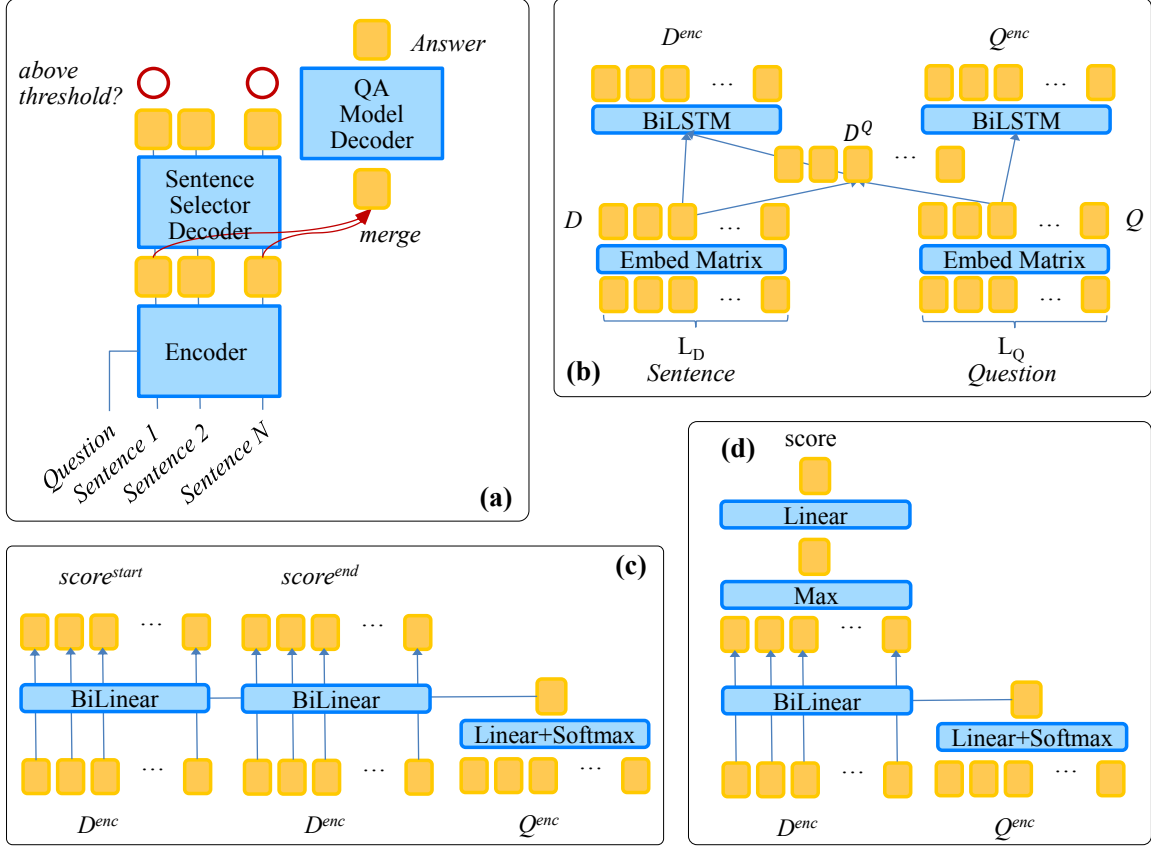
Figure 2: Our model architecture. (a) Overall pipeline, consisting of sentence selector and QA model. Selection score of each sentence is obtained in parallel, then sentences with selection score above the threshold are merged and fed into QA model. (b) Shared encoder of sentence selector and S-Reader (QA Model), which takes document and the question as inputs and outputs the document encodings D^{enc} and question encodings Q^{enc}. (c) Decoder of S-Reader (QA Model), which takes D^{enc} and Q^{enc} as inputs and outputs the scores for start and end positions. (d) Decoder of sentence selector, which takes D^{enc} and Q^{enc} for each sentence and outputs the score indicating if the question is answerable given the sentence.

a trainable weight matrix. After this, sentence encodings and question encodings are obtained using an LSTM (Hochreiter and Schmidhuber, 1997).

$$D^{enc} = \text{BiLSTM}([D_i; D_i^q]) \in \mathbb{R}^{h \times L_d} \quad (3)$$

$$Q^{enc} = \text{BiLSTM}(Q_j) \in \mathbb{R}^{h \times L_q} \quad (4)$$

Here, ';' denotes the concatenation of two vectors, and h is a hyperparameter of the hidden dimension.

Next, the decoder is a task-specific module which computes the score for the sentence by calculating bilinear similarities between sentence encodings and question encodings as follows.

$$\beta = \text{softmax}(w^T Q^{enc}) \in \mathbb{R}^{L_q} \quad (5)$$

$$q^{\tilde{enc}} = \sum_{j=1}^{L_q} (\beta_j Q_j^{enc}) \in \mathbb{R}^h \quad (6)$$

$$\tilde{h}_i = (D_i^{enc} W_2 q^{\tilde{enc}}) \in \mathbb{R}^h \quad (7)$$

$$\tilde{h} = \max(\tilde{h_1}, \tilde{h_2}, \cdots, \tilde{h_{L_d}}) \quad (8)$$

$$score = W_3^T \tilde{h} \in \mathbb{R}^2 \quad (9)$$

Here, $w \in \mathbb{R}^h, W_2 \in \mathbb{R}^{h \times h \times h}, W_3 \in \mathbb{R}^{h \times 2}$, are trainable weight matrices. Each dimension in *score* means the question is answerable or nonanswerable given the sentence.

We introduce 3 techniques to train the model. (i) As the encoder module of our model is identical to that of S-Reader, we transfer the weights to the encoder module from the QA model trained on the single oracle sentence (ORACLE). (ii) We modify the training data by treating a sentence as a wrong sentence if the QA model gets 0 F1, even if the sentence is the oracle sentence. (iii) After we

Dataset	Domain	N word	N sent	N doc	Supervision
SQuAD	Wikipedia	155	5	-	Span
NewsQA	News Articles	803	20	-	Span
TriviaQA (Wikipedia)	Wikipedia	11202	488	2	Distant
SQuAD-Open	Wikipedia	120734	4488	10	Distant
SQuAD-Adversarial-AddSent	Wikipedia	169	6	-	Span
SQuAD-Adversarial-AddOneSent	Wikipedia	165	6	-	Span

Table 3: Dataset used for experiments. 'N word', 'N sent' and 'N doc' refer to the average number of words, sentences and documents, respectively. All statistics are calculated on the development set. For SQuAD-Open, since the task is in open-domain, we calculated the statistics based on top 10 documents from Document Retriever in DrQA (Chen et al., 2017).

obtain the score for each sentence, we normalize scores across sentences from the same paragraph, similar to Clark and Gardner (2017). All of these three techniques give substantial improvements in sentence selection accuracy, as shown in Table 4. More details including hyperparameters and training procedures are shown in Appendix A.

Because the minimal set of sentences required to answer the question depends on the question, we select the set of sentences by thresholding the sentence scores, where the threshold is a hyperparameter (details in Appendix A). This method allows the model to select a variable number of sentences for each question, as opposed to a fixed number of sentences for all questions. Also, by controlling the threshold, the number of sentences can be dynamically controlled during the inference. We define `Dyn` (for Dynamic) as this method, and define `Top k` as the method which simply selects the top-k sentences for each question.

4 Experiments

4.1 Dataset and Evaluation Metrics

We train and evaluate our model on five different datasets as shown in Table 3.

SQuAD (Rajpurkar et al., 2016) is a well-studied QA dataset on Wikipedia articles that requires each question to be answered from a paragraph.

NewsQA (Trischler et al., 2016) is a dataset on news articles that also provides a paragraph for each question, but the paragraphs are longer than those in SQuAD.

TriviaQA (Joshi et al., 2017) is a dataset on a large set of documents from the Wikipedia domain and Web domain. Here, we only use the Wikipedia

domain. Each question is given a much longer context in the form of multiple documents.

SQuAD-Open (Chen et al., 2017) is an open-domain question answering dataset based on SQuAD. In SQuAD-Open, only the question and the answer are given. The model is responsible for identifying the relevant context from all English Wikipedia articles.

SQuAD-Adversarial (Jia and Liang, 2017) is a variant of SQuAD. It shares the same training set as SQuAD, but an adversarial sentence is added to each paragraph in a subset of the development set.

We use accuracy (Acc) and mean average precision (MAP) to evaluate sentence selection. We also measure the average number of selected sentences (N sent) to compare the efficiency of our `Dyn` method and the `Top k` method.

To evaluate the performance in the task of question answering, we measure F1 and EM (Exact Match), both being standard metrics for evaluating span-based QA. In addition, we measure training speed (Train Sp) and inference speed (Infer Sp) relative to the speed of standard QA model (FULL). The speed is measured using a single GPU (Tesla K80), and includes the training and inference time for the sentence selector.

4.2 SQuAD and NewsQA

For each QA model, we experiment with three types of inputs. First, we use the full document (FULL). Next, we give the model the oracle sentence containing the groundtruth answer span (ORACLE). Finally, we select sentences using our sentence selector (MINIMAL), using both `Top k` and `Dyn`. We also compare this last method with TF-IDF method for sentence selection, which selects sentences using n-gram TF-IDF distance between each sentence and the question.

Model	SQuAD		NewsQA		
	Top 1	MAP	Top 1	Top 3	MAP
TF-IDF	81.2	89.0	49.8	72.1	63.7
Our selector	85.8	91.6	63.2	85.1	75.5
Our selector (T)	90.0	94.3	67.1	87.9	78.5
Our selector (T+M, T+M+N)	**91.2**	**95.0**	**70.9**	**89.7**	**81.1**
Tan et al. (2018)	-	92.1	-	-	-

Selection method	SQuAD		NewsQA	
	N sent	Acc	N sent	Acc
Top k (T+M)[a]	1	91.2	1	70.9
Top k (T+M)[a]	2	97.2	3	89.7
Top k (T+M)[a]	3	98.9	4	92.5
Dyn (T+M)	1.5	94.7	2.9	84.9
Dyn (T+M)	1.9	96.5	3.9	89.4
Dyn (T+M+N)	1.5	98.3	2.9	91.8
Dyn (T+M+N)	1.9	**99.3**	3.9	**94.6**

Table 4: Results of sentence selection on the dev set of SQuAD and NewsQA. (Top) We compare different models and training methods. We report Top 1 accuracy (Top 1) and Mean Average Precision (MAP). Our selector outperforms the previous state-of-the-art (Tan et al., 2018). (Bottom) We compare different selection methods. We report the number of selected sentences (N sent) and the accuracy of sentence selection (Acc). 'T', 'M' and 'N' are training techniques described in Section 3.2 (weight transfer, data modification and score normalization, respectively).

[a] 'N' does not change the result on Top k, since Top k depends on the relative scores across the sentences from same paragraph.
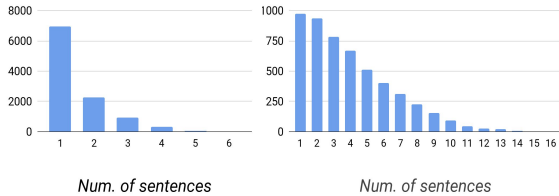

Figure 3: The distributions of number of sentences that our selector selects using Dyn method on the dev set of SQuAD (left) and NewsQA (right).

Results Table 4 shows results in the task of sentence selection on SQuAD and NewsQA. First, our selector outperforms TF-IDF method and the previous state-of-the-art by large margin (up to 2.9% MAP).

Second, our three training techniques – weight transfer, data modification and score normalization – improve performance by up to 5.6% MAP. Finally, our Dyn method achieves higher accuracy with less sentences than the Top k method. For example, on SQuAD, Top 2 achieves 97.2 accuracy, whereas Dyn achieves 99.3 accuracy with

SQuAD (with S-Reader)				
	F1	EM	Train Sp	Infer Sp
FULL	79.9	71.0	x1.0	x1.0
ORACLE	84.3	74.9	x6.7	x5.1
MINIMAL(Top k)	78.7	69.9	**x6.7**	**x5.1**
MINIMAL(Dyn)	79.8	70.9	**x6.7**	x3.6
SQuAD (with DCN+)				
FULL	83.1	74.5	x1.0	x1.0
ORACLE	85.1	76.0	x3.0	x5.1
MINIMAL(Top k)	79.2	70.7	x3.0	**x5.1**
MINIMAL(Dyn)	80.6	72.0	x3.0	x3.7
GNR	75.0[a]	66.6[a]	-	-
FastQA	78.5	70.3	-	-
FusionNet	**83.6**	**75.3**	-	-
NewsQA (with S-Reader)				
	F1	EM	Train Sp	Infer Sp
FULL	63.8	50.7	x1.0	x1.0
ORACLE	75.5	59.2	x18.8	x21.7
MINIMAL(Top k)	62.3	49.3	**x15.0**	**x6.9**
MINIMAL(Dyn)	**63.2**	**50.1**	**x15.0**	x5.3
FastQA	56.1	43.7	-	-

Table 5: Results on the dev set of SQuAD (First two) and NewsQA (Last). For Top k, we use $k = 1$ and $k = 3$ for SQuAD and NewsQA, respectively. We compare with GNR (Raiman and Miller, 2017), FusionNet (Huang et al., 2018) and FastQA (Weissenborn et al., 2017), which are the model leveraging sentence selection for question answering, and the published state-of-the-art models on SQuAD and NewsQA, respectively.

[a] Numbers on the test set.

1.9 sentences per example. On NewsQA, Top 4 achieves 92.5 accuracy, whereas Dyn achieves 94.6 accuracy with 3.9 sentences per example.

Figure 3 shows that the number of sentences selected by Dyn method vary substantially on both SQuAD and NewsQA. This shows that Dyn chooses a different number of sentences depending on the question, which reflects our intuition.

Table 5 shows results in the task of QA on SQuAD and NewsQA. MINIMAL is more efficient in training and inference than FULL. On SQuAD, S-Reader achieves 6.7× training and 3.6× inference speedup on SQuAD, and 15.0× training and 6.9× inference speedup on NewsQA. In addition to the speedup, MINIMAL achieves comparable result to FULL (using S-Reader, 79.9 vs 79.8 F1 on SQuAD and 63.8 vs 63.2 F1 on NewsQA).

We compare the predictions from FULL and MINIMAL in Table 6. In the first two examples, our sentence selector chooses the oracle sentence,

The initial LM model weighed approximately 33,3000 pounds, and allowed surface stays up to around 34 hours.	
. . .	
An Extended Lunar Module weighed over 36,200 pounds, and allowed surface stays of over <u>3 days</u>. ✓	
For about how long would the extended LM allow a surface stay on the moon?	

Approximately <u>1,000</u> British soldiers were killed or injured. ✓	
. . .	
The remaining 500 British troops, led by George Washington, retreated to Virginia.	
How many casualties did British get?	

This book, which influenced the thought of Charles Darwin, successfully promoted the doctrine of <u>uniformitarianism</u>.	
This theory states that slow geological processes have occurred throughout the Earth's history and are still occurring today. ✓	
In contrast, catastrophism is the theory that Earth's features formed in single, catastrophic events and remained unchanged thereafter. ✓	
Which theory states that slow geological processes are still occuring today, and have occurred throughout Earth's history?	

Table 6: Examples on SQuAD. Grountruth span (<u>underlined text</u>), the prediction from FULL (blue text) and MINIMAL (red text). Sentences selected by our selector is denoted with ✓. In the above two examples, MINIMAL correctly answer the question by selecting the oracle sentence. In the last example, MINIMAL fails to answer the question, since the inference over first and second sentences is required to answer the question.

selected	sentence
✓ ✓ ✓	However, in 1883-84 Germany began to build a colonial empire in Africa and the South Pacific, before losing interest in imperialism.
✓ ✓	The establishment of the German colonial empire proceeded smoothly, starting with German New Guinea in <u>1884</u>.
When did Germany found their first settlement? 1883-84 1884 1884	
✓ ✓ ✓	In the <u>late 1920s</u>, Tesla also befriended George Sylvester Viereck, a poet, writer, mystic, and later, a Nazi propagandist.
✓	In middle age, Tesla became a close friend of Mark Twain; they spent a lot of time together in his lab and elsewhere.
When did Tesla become friends with Viereck? late 1920s middle age late 1920s	

Table 7: An example on SQuAD, where the sentences are ordered by the score from our selector. Grountruth span (<u>underlined text</u>), the predictions from `Top 1` (blue text), `Top 2` (green text) and `Dyn` (red text). Sentences selected by `Top 1`, `Top 2` and `Dyn` are denoted with ✓, ✓ and ✓, respectively.

and the QA model correctly answers the question. In the last example, our sentence selector fails to choose the oracle sentence, so the QA model cannot predict the correct answer. In this case, our selector chooses the second and the third sentences instead of the oracle sentence because the former contains more information relevant to question. In fact, the context over the first and the second sentences is required to correctly answer the question.

Table 7 shows an example on SQuAD, which MINIMAL with `Dyn` correctly answers the question, and MINIMAL with `Top k` sometimes does not. `Top 1` selects one sentence in the first example, thus fails to choose the oracle sentence. `Top 2` selects two sentences in the second example, which is inefficient as well as leads to the wrong answer. In both examples, `Dyn` selects the oracle sentence with minimum number of sentences, and subsequently predicts the answer. More analyses are shown in Appendix B.

4.3 TriviaQA and SQuAD-Open

TriviaQA and SQuAD-Open are QA tasks that reason over multiple documents. They do not provide the answer span and only provide the question-answer pairs.

For each QA model, we experiment with two types of inputs. First, since TriviaQA and SQuAD-Open have many documents for each question, we first filter paragraphs based on the TF-IDF similarities between the question and the paragraph, and then feed the full paragraphs to the QA model (FULL). On TriviaQA, we choose the top 10 paragraphs for training and inference. On SQuAD-Open, we choose the top 20 paragraphs for training and the top 40 for inferences. Next, we use our sentence selector with `Dyn` (MINIMAL). We select 5-20 sentences using our sentence selector, from 200 sentences based on TF-IDF.

For training the sentence selector, we use two techniques described in Section 3.2, weight transfer and score normalization, but we do not use data modification technique, since there are too many sentences to feed each of them to the QA model. For training the QA model, we transfer the weights from the QA model trained on SQuAD, then fine-tune.

		TriviaQA (Wikipedia)					SQuAD-Open				
		n sent	Acc	Sp	F1	EM	n sent	Acc	Sp	F1	EM
FULL		69	95.9	x1.0	59.6	53.5	124	76.9	x1.0	41.0	33.1
MINIMAL	TF-IDF	5	73.0	x13.8	51.9	45.8	5	46.1	x12.4	36.6	29.6
		10	79.9	x6.9	57.2	51.5	10	54.3	x6.2	39.8	32.5
	Our Selector	5.0	84.9	**x13.8**	59.5	54.0	5.3	58.9	**x11.7**	42.3	**34.6**
		10.5	90.9	x6.6	**60.5**	**54.9**	10.7	64.0	x5.8	**42.5**	**34.7**
Rank 1		-	-	-	56.0[a]	51.6[a]	2376[a]	77.8	-	-	29.8
Rank 2		-	-	-	55.1[a]	48.6[a]	-	-	-	37.5	29.1
Rank 3		-	-	-	52.9[b]	46.9[a]	2376[a]	77.8	-	-	28.4

Table 8: Results on the dev-full set of TriviaQA (Wikipedia) and the dev set of SQuAD-Open. Full results (including the dev-verified set on TriviaQA) are in Appendix C. For training FULL and MINIMAL on TriviaQA, we use 10 paragraphs and 20 sentences, respectively. For training FULL and MINIMAL on SQuAD-Open, we use 20 paragraphs and 20 sentences, respectively. For evaluating FULL and MINIMAL, we use 40 paragraphs and 5-20 sentences, respectively. 'n sent' indicates the number of sentences used during inference. 'Acc' indicates accuracy of whether answer text is contained in selected context. 'Sp' indicates inference speed. We compare with the results from the sentences selected by TF-IDF method and our selector (`Dyn`). We also compare with published Rank1-3 models. For TriviaQA(Wikipedia), they are Neural Casecades (Swayamdipta et al., 2018), Reading Twice for Natural Language Understanding (Weissenborn, 2017) and Mnemonic Reader (Hu et al., 2017). For SQuAD-Open, they are DrQA (Chen et al., 2017) (Multitask), R[3] (Wang et al., 2018) and DrQA (Plain).

_____

[a]Approximated based on there are 475.2 sentences per document, and they use 5 documents per question

[b]Numbers on the test set.

Results Table 8 shows results on TriviaQA (Wikipedia) and SQuAD-Open. First, MINIMAL obtains higher F1 and EM over FULL, with the inference speedup of up to 13.8×. Second, the model with our sentence selector with `Dyn` achieves higher F1 and EM over the model with TF-IDF selector. For example, on the development-full set, with 5 sentences per question on average, the model with `Dyn` achieves 59.5 F1 while the model with TF-IDF method achieves 51.9 F1. Third, we outperforms the published state-of-the-art on both dataset.

4.4 SQuAD-Adversarial

We use the same settings as Section 4.2. We use the model trained on SQuAD, which is exactly same as the model used for Table 5. For MINIMAL, we select top 1 sentence from our sentence selector to the QA model.

Results Table 9 shows that MINIMAL outperforms FULL, achieving the new state-of-the-art by large margin (+11.1 and +11.5 F1 on AddSent and AddOneSent, respectively).

Figure 10 compares the predictions by DCN+ FULL (blue) and MINIMAL (red). While FULL selects the answer from the adversarial sentence, MINIMAL first chooses the oracle sentence, and

SQuAD-Adversarial		AddSent			AddOneSent		
		F1	EM	Sp	F1	EM	Sp
DCN+	FULL	52.6	46.2	x0.7	63.5	56.8	x0.7
	ORACLE	84.2	75.3	x4.3	84.5	75.8	x4.3
	MINIMAL	**59.7**	**52.2**	x4.3	**67.5**	**60.1**	x4.3
S-Reader	FULL	57.7	51.1	x1.0	66.5	59.7	x1.0
	ORACLE	82.5	74.1	x6.0	82.9	74.6	x6.0
	MINIMAL	58.5	51.5	**x6.0**	66.5	59.5	**x6.0**
RaSOR		39.5	-	-	49.5	-	-
ReasoNet		39.4	-	-	50.3	-	-
Mnemonic Reader		46.6	-	-	56.0	-	-

Table 9: Results on the dev set of SQuAD-Adversarial. We compare with RaSOR (Lee et al., 2016), ReasoNet (Shen et al., 2017) and Mnemonic Reader (Hu et al., 2017), the previous state-of-the-art on SQuAD-Adversarial, where the numbers are from Jia and Liang (2017).

subsequently predicts the correct answer. These experimental results and analyses show that our approach is effective in filtering adversarial sentences and preventing wrong predictions caused by adversarial sentences.

5 Related Work

Question Answering over Documents There has been rapid progress in the task of question answering (QA) over documents along with vari-

San Francisco mayor <u>Ed Lee</u> said of the highly visible homeless presence in this area "they are going to have to leave".   Jeff Dean was the mayor of Diego Diego during Champ Bowl 40.
Who was the mayor of San Francisco during Super Bowl 50?
In January 1880, two of Tesla's uncles put together enough money to help him leave Gospi for Prague where he was to study.   Tadakatsu moved to the city of Chicago in 1881.
What city did Tesla move to in 1880?

Table 10: Examples on SQuAD-Adversarial. Groundtruth span is in <u>underlined text</u>, and predictions from FULL and MINIMAL are in blue text and red text, respectively.

ous datasets and competitive approaches. Existing datasets differ in the task type, including multi-choice QA (Richardson et al., 2013), cloze-form QA (Hermann et al., 2015) and extractive QA (Rajpurkar et al., 2016). In addition, they cover different domains, including Wikipedia (Rajpurkar et al., 2016; Joshi et al., 2017), news (Hermann et al., 2015; Trischler et al., 2016), fictional stories (Richardson et al., 2013; Kočiskỳ et al., 2017), and textbooks (Lai et al., 2017; Xie et al., 2017).

Many neural QA models have successfully addressed these tasks by leveraging coattention or bidirectional attention mechanisms (Xiong et al., 2018; Seo et al., 2017) to model the codependent context over the document and the question. However, Jia and Liang (2017) find that many QA models are sensitive to adversarial inputs.

Recently, researchers have developed large-scale QA datasets, which requires answering the question over a large set of documents in a closed (Joshi et al., 2017) or open-domain (Dunn et al., 2017; Berant et al., 2013; Chen et al., 2017; Dhingra et al., 2017). Many models for these datasets either retrieve documents/paragraphs relevant to the question (Chen et al., 2017; Clark and Gardner, 2017; Wang et al., 2018), or leverage simple non-recurrent architectures to make training and inference tractable over large corpora (Swayamdipta et al., 2018; Yu et al., 2018).

Sentence selection The task of selecting sentences that can answer to the question has been studied across several QA datasets (Yang et al., 2015), by modeling relevance between a sentence and the question (Yin et al., 2016; Miller et al., 2016; Min et al., 2017). Several recent works also study joint sentence selection and question answering. Choi et al. (2017) propose a framework that identifies the sentences relevant to the question (property) using simple bag-of-words representation, then generates the answer from those sentences using recurrent neural networks. Raiman and Miller (2017) cast the task of

extractive question answering as a search problem by iteratively selecting the sentences, start position and end position. They are different from our work in that (i) we study of the minimal context required to answer the question, (ii) we choose the minimal context by selecting variable number of sentences for each question, while they use a fixed size of number as a hyperparameter, (iii) our framework is flexible in that it does not require end-to-end training and can be combined with existing QA models, and (iv) they do not show robustness to adversarial inputs.

6 Conclusion

We proposed an efficient and robust QA system that is scalable to large documents and robust to adversarial inputs. First, we studied the minimal context required to answer the question in existing datasets and found that most questions can be answered using a small set of sentences. Second, inspired by this observation, we proposed a sentence selector which selects a minimal set of sentences to answer the question to give to the QA model. We demonstrated the efficiency and effectiveness of our method across five different datasets with varying sizes of source documents. We achieved the training and inference speedup of up to $15\times$ and $13\times$, respectively, and accuracy comparable to or better than existing state-of-the-art. In addition, we showed that our approach is more robust to adversarial inputs.

Acknowledgments

References

Jonathan Berant, Andrew Chou, Roy Frostig, and Percy Liang. 2013. Semantic parsing on freebase from question-answer pairs. In *EMNLP*.

Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. 2017. Reading wikipedia to answer open-domain questions. In *ACL*.

Eunsol Choi, Daniel Hewlett, Jakob Uszkoreit, Illia Polosukhin, Alexandre Lacoste, and Jonathan Berant. 2017. Coarse-to-fine question answering for long documents. In *ACL*.

Christopher Clark and Matt Gardner. 2017. Simple and effective multi-paragraph reading comprehension. *arXiv preprint arXiv:1710.10723* .

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, 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* .

Kazuma Hashimoto, Caiming Xiong, Yoshimasa Tsuruoka, and Richard Socher. 2017. A joint many-task model: Growing a neural network for multiple nlp tasks. In *EMNLP*.

Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In *Advances in Neural Information Processing Systems*.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation* .

Minghao Hu, Yuxing Peng, and Xipeng Qiu. 2017. Mnemonic reader for machine comprehension. *arXiv preprint arXiv:1705.02798* .

Hsin-Yuan Huang, Chenguang Zhu, Yelong Shen, and Weizhu Chen. 2018. Fusionnet: Fusing via fully-aware attention with application to machine comprehension. In *ICLR*.

Robin Jia and Percy Liang. 2017. Adversarial examples for evaluating reading comprehension systems. In *EMNLP*.

Mandar Joshi, Eunsol Choi, Daniel S Weld, and Luke Zettlemoyer. 2017. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. In *ACL*.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1706.02596v2* .

Tomáš Kočiskỳ, Jonathan Schwarz, Phil Blunsom, Chris Dyer, Karl Moritz Hermann, Gábor Melis, and Edward Grefenstette. 2017. The narrativeqa reading comprehension challenge. *arXiv preprint arXiv:1712.07040* .

Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard Hovy. 2017. Race: Large-scale reading comprehension dataset from examinations. In *EMNLP*.

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* .

Christopher Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven Bethard, and David McClosky. 2014. The stanford corenlp natural language processing toolkit. In *ACL*.

Bryan McCann, James Bradbury, Caiming Xiong, and Richard Socher. 2017. Learned in translation: Contextualized word vectors. In *NIPS*.

Alexander Miller, Adam Fisch, Jesse Dodge, Amir-Hossein Karimi, Antoine Bordes, and Jason Weston. 2016. Key-value memory networks for directly reading documents. In *EMNLP*.

Sewon Min, Minjoon Seo, and Hannaneh Hajishirzi. 2017. Question answering through transfer learning from large fine-grained supervision data. In *ACL*.

Boyuan Pan, Hao Li, Zhou Zhao, Bin Cao, Deng Cai, and Xiaofei He. 2017. Memen: Multi-layer embedding with memory networks for machine comprehension. *arXiv preprint arXiv:1707.09098* .

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *EMNLP*.

Jonathan Raiman and John Miller. 2017. Globally normalized reader. In *EMNLP*.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. In *EMNLP*.

Matthew Richardson, Christopher JC Burges, and Erin Renshaw. 2013. Mctest: A challenge dataset for the open-domain machine comprehension of text. In *EMNLP*.

Minjoon Seo, Aniruddha Kembhavi, Ali Farhadi, and Hannaneh Hajishirzi. 2017. Bidirectional attention flow for machine comprehension. In *ICLR*.

Yelong Shen, Po-Sen Huang, Jianfeng Gao, and Weizhu Chen. 2017. Reasonet: Learning to stop reading in machine comprehension. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*.

Nitish Srivastava, Geoffrey E Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *Journal of machine learning research* .

Swabha Swayamdipta, Ankur P Parikh, and Tom Kwiatkowski. 2018. Multi-mention learning for reading comprehension with neural cascades. In *ICLR*.

Chuanqi Tan, Furu Wei, Qingyu Zhou, Nan Yang, Bowen Du, Weifeng Lv, and Ming Zhou. 2018. Context-aware answer sentence selection with hierarchical gated recurrent neural networks. *IEEE/ACM Transactions on Audio, Speech, and Language Processing* .

Adam Trischler, Tong Wang, Xingdi Yuan, Justin Harris, Alessandro Sordoni, Philip Bachman, and Kaheer Suleman. 2016. Newsqa: A machine comprehension dataset. *arXiv preprint arXiv:1611.09830* .

Shuohang Wang, Mo Yu, Xiaoxiao Guo, Zhiguo Wang, Tim Klinger, Wei Zhang, Shiyu Chang, Gerald Tesauro, Bowen Zhou, and Jing Jiang. 2018. R3: Reinforced reader-ranker for open-domain question answering. In *AAAI*.

Dirk Weissenborn. 2017. Reading twice for natural language understanding. *CoRR* abs/1706.02596.

Dirk Weissenborn, Georg Wiese, and Laura Seiffe. 2017. Making neural qa as simple as possible but not simpler. In *CoNLL*.

Qizhe Xie, Guokun Lai, Zihang Dai, and Eduard Hovy. 2017. Large-scale cloze test dataset designed by teachers. *arXiv preprint arXiv:1711.03225* .

Caiming Xiong, Victor Zhong, and Richard Socher. 2018. Dcn+: Mixed objective and deep residual coattention for question answering. In *ICLR*.

Yi Yang, Wen-tau Yih, and Christopher Meek. 2015. Wikiqa: A challenge dataset for open-domain question answering. In *EMNLP*.

Wenpeng Yin, Hinrich Schtze, Bing Xiang, and Bowen Zhou. 2016. Abcnn: Attention-based convolutional neural network for modeling sentence pairs. *TACL* .

Adams Wei Yu, David Dohan, Quoc Le, Thang Luong, Rui Zhao, and Kai Chen. 2018. Fast and accurate reading comprehension by combining self-attention and convolution. In *ICLR*.