WIKIQA: A Challenge Dataset for Open-Domain Question Answering

Yi Yang*
Georgia Institute of Technology
Atlanta, GA 30308, USA
yiyang@gatech.edu

Wen-tau Yih **Christopher Meek**
Microsoft Research
Redmond, WA 98052, USA
{scottyih,meek}@microsoft.com

Abstract

We describe the WIKIQA dataset, a new publicly available set of question and sentence pairs, collected and annotated for research on open-domain question answering. Most previous work on answer sentence selection focuses on a dataset created using the TREC-QA data, which includes editor-generated questions and candidate answer sentences selected by matching content words in the question. WIKIQA is constructed using a more natural process and is more than an order of magnitude larger than the previous dataset. In addition, the WIKIQA dataset also includes questions for which there are no correct sentences, enabling researchers to work on answer triggering, a critical component in any QA system. We compare several systems on the task of answer sentence selection on both datasets and also describe the performance of a system on the problem of answer triggering using the WIKIQA dataset.

1 Introduction

Answer sentence selection is a crucial subtask of the open-domain question answering (QA) problem, with the goal of extracting answers from a set of pre-selected sentences (Heilman and Smith, 2010; Yao et al., 2013; Severyn and Moschitti, 2013). In order to conduct research on this important problem, Wang et al. (2007) created a dataset, which we refer to by QASENT, based on the TREC-QA data. The QASENT dataset chose questions in TREC 8-13 QA tracks and selected sentences that share one or more non-stopwords from the questions. Although QASENT has since become the benchmark dataset for the answer selection problem, its creation process actually introduces a strong bias in the types of answers that are included. The following example illustrates an answer that does not share any content words with the question and would not be selected:

Q: *How did Seminole war end?*
A: *Ultimately, the Spanish Crown ceded the colony to United States rule.*

One significant concern with this approach is that the lexical overlap will make sentence selection easier for the QASENT dataset and might inflate the performance of existing systems in more natural settings. For instance, Yih et al. (2013) find that simple word matching methods outperform many sophisticated approaches on the dataset. We explore this possibility in Section 3.

A second, more subtle challenge for question answering is that it normally assumes that there is at least one correct answer for each question in the candidate sentences. During the data construction procedures, all the questions without correct answers are manually discarded.[1] We address a new challenge of *answer triggering*, an important component in QA systems, where the goal is to detect whether there exist correct answers in the set of candidate sentences for the question, and return a correct answer if there exists such one.

We present WIKIQA, a dataset for open-domain question answering.[2] The dataset contains 3,047 questions originally sampled from Bing query logs. Based on the user clicks, each question is associated with a Wikipedia page presumed to be the topic of the question. In order to eliminate answer sentence biases caused by keyword matching, we consider all the sentences in

*Work conducted while interning at Microsoft Research.

[1] The policy is adopted both by the official QASENT tracks (Voorhees and Tice, 1999) and by Wang et al. (2007).

[2] The data and evaluation script can be downloaded at http://aka.ms/WikiQA.

the summary paragraph of the page as the candidate answer sentences, with labels on whether the sentence is a correct answer to the question provided by crowdsourcing workers. Among these questions, about one-third of them contain correct answers in the answer sentence set.

We implement several strong baselines to study model behaviors in the two datasets, including two previous state-of-the-art systems (Yih et al., 2013; Yu et al., 2014) on the QASENT dataset as well as simple lexical matching methods. The results show that lexical semantic methods yield better performance than sentence semantic models on QASENT, while sentence semantic approaches (e.g., convolutional neural networks) outperform lexical semantic models on WIKIQA. We propose to evaluate answer triggering using question-level precision, recall and F_1 scores. The best F_1 scores are slightly above 30%, which suggests a large room for improvement.

2 WIKIQA Dataset

In this section, we describe the process of creating our WIKIQA dataset in detail, as well as some comparisons to the QASENT dataset.

2.1 Question & Sentence Selection

In order to reflect the true information need of general users, we used Bing query logs as the question source. Taking the logs from the period of May 1st, 2010 to July 31st, 2011, we first selected question-like queries using simple heuristics, such as queries starting with a WH-word (e.g., "what" or "how") and queries ending with a question mark. In addition, we filtered out some entity queries that satisfy the rules, such as the TV show "how I met your mother." In the end, approximately 2% of the queries were selected. To focus on factoid questions and to improve the question quality, we then selected only the queries issued by at least 5 unique users and have clicks to Wikipedia. Among them, we sampled 3,050 questions based on query frequencies.

Because the summary section of a Wikipedia page provides the basic and usually most important information about the topic, we used sentences in this section as the candidate answers. Figure 1 shows an example question, as well as the summary section of a linked Wikipedia page.

Question: Who wrote second Corinthians?

Second Epistle to the Corinthians The Second Epistle to the Corinthians, often referred to as Second Corinthians (and written as 2 Corinthians), is the eighth book of the New Testament of the Bible. Paul the Apostle and "Timothy our brother" wrote this epistle to "the church of God which is at Corinth, with all the saints which are in all Achaia".

Figure 1: An example question and the summary paragraph of a Wikipedia page.

2.2 Sentence Annotation

We employed crowdsourcing workers through a platform, which is similar to Amazon MTurk, to label whether the candidate answer sentences of a question are correct. We designed a cascaded Web UI that consists of two stages. The first stage shows a testing question, along with the title and the summary paragraph of the associated Wikipedia page, asking the worker "Does the short paragraph answer the question?" If the worker chooses "No", then equivalently all the sentences in this paragraph are marked incorrect and the UI moves to the next question. Otherwise, the system enters the second stage and puts a checkbox along each sentence. The worker is then asked to check the sentences that can answer the question *in isolation*, assuming coreference is resolved. To ensure the label quality, each question was labeled by three workers. Sentences with inconsistent labels would be verified by a different set of crowdsourcing workers. The final decision was based on the majority vote of all the workers. In the end, we included 3,047 questions and 29,258 sentences in the dataset, where 1,473 sentences were labeled as answer sentences to their corresponding questions.

Although not used in the experiments, each of these answer sentence is associated with the *answer phrase*, which is defined as the shortest substring of the sentence that answers the question. For instance, the second sentence in the summary paragraph shown in Figure 1 is an answer sentence. Its substring "Paul the Apostle and Timothy our brother" can be treated as the answer phrase. The annotations of the answer phrases were given by the authors of this paper. Because the answer phrase boundary can be highly ambiguous, each sentence is associated with at most two answer phrases that are both acceptable, given by two different labelers. We hope this addition to the WIKIQA data can be beneficial to future researchers for building or evaluating an end-to-end question answering system.

	Train	Dev	Test	Total
# of ques.	94	65	68	227
# of sent.	5,919	1,117	1,442	8,478
# of ans.	475	205	248	928
Avg. len. of ques.	11.39	8.00	8.63	9.59
Avg. len. of sent.	30.39	24.90	25.61	28.85

Table 1: Statistics of the QASENT dataset.

	Train	Dev	Test	Total
# of ques.	2,118	296	633	3,047
# of sent.	20,360	2,733	6,165	29,258
# of ans.	1,040	140	293	1,473
Avg. len. of ques.	7.16	7.23	7.26	7.18
Avg. len. of sent.	25.29	24.59	24.95	25.15
# of ques. w/o ans.	1,245	170	390	1,805

Table 2: Statistics of the WIKIQA dataset.

2.3 WIKIQA vs. QASENT

Our WIKIQA dataset differs from the existing QASENT dataset in both question and candidate answer sentence distributions. Questions in QASENT were originally used in TREC 8-13 QA tracks and were a mixture of questions from query logs (e.g., Excite and Encarta) and from human editors. The questions might be outdated and do not reflect the true information need of a QA system user. By contrast, questions in WIKIQA were sampled from real queries of Bing without editorial revision. On the sentence side, the candidate sentences in QASENT were selected from documents returned by past participating teams in the TREC QA tracks, and sentences were only included if they shared content words from the questions. These procedures make the distribution of the candidate sentence skewed and unnatural. In comparison, 20.3% of the answers in the WIKIQA dataset share no content words with questions. Candidate sentences in WIKIQA were chosen from relevant Wikipedia pages directly, which could be closer to the input of an answer sentence selection module of a QA system.

To make it easy to compare results of different QA systems when evaluated on the WIKIQA dataset, we randomly split the data to training (70%), development (10%) and testing (20%) sets. Some statistics of the QASENT and WIKIQA datasets are presented in Tables 1 and 2.[3] WIKIQA contains an order of magnitude more questions and three times more answer sentences compared to QASENT. Unlike QASENT, we did not filter questions with only incorrect answers, as they are still valuable for model training and more importantly, useful for evaluating the task of answer triggering, as described in Section 3. Specifically, we find nearly two-thirds of questions contain no correct answers in the candidate sentences.

Class	QASENT	WIKIQA
Location	37 (16%)	373 (12%)
Human	65 (29%)	494 (16%)
Numeric	70 (31%)	658 (22%)
Abbreviation	2 (1%)	16 (1%)
Entity	37 (16%)	419 (14%)
Description	16 (7%)	1087 (36%)

Table 3: Question classes of the QASENT and WIKIQA datasets.

The distributions of question types in these two datasets are also different, as shown in Table 3.[4] WIKIQA contains more description or definition questions, which could be harder to answer.

3 Experiments

Many systems have been proposed and tested on the QASENT dataset, including lexical semantic models (Yih et al., 2013) and sentence semantic models (Yu et al., 2014). We investigate the performance of several systems on WIKIQA and QASENT. As discussed in Section 2, WIKIQA offers us the opportunity to evaluate QA systems on answer triggering. We propose simple metrics and perform a feature study on the new task. Finally, we include some error analysis and discussion at the end of this section.

3.1 Baseline Systems

We consider two simple word matching methods: Word Count and Weighted Word Count. The first method counts the number of non-stopwords in the question that also occur in the answer sentence. The second method re-weights the counts by the IDF values of the question words.

We reimplement LCLR (Yih et al., 2013), an answer sentence selection approach that achieves very competitive results on QASENT. LCLR

[3] We follow experimental settings of Yih et al. (2013) on the QASENT dataset. Although the training set in the original data contains more questions, only 94 of them are paired with sentences that have human annotations.

[4] The classifier is trained using a logistic regression model on the UIUC Question Classification Datasets (`http://cogcomp.cs.illinois.edu/Data/QA/QC`). The performance is comparable to (Li and Roth, 2002).

Model	QASENT		WIKIQA	
	MAP	MRR	MAP	MRR
Word Cnt	0.5919	0.6662	0.4891	0.4924
Wgt Word Cnt	0.6095	0.6746	0.5099	0.5132
LCLR	**0.6954**	0.7617	0.5993	0.6086
PV	0.5213	0.6023	0.5110	0.5160
CNN	0.5590	0.6230	0.6190	0.6281
PV-Cnt	0.6762	0.7514	0.5976	0.6058
CNN-Cnt	0.6951	**0.7633**	**0.6520**	**0.6652**

Table 4: Baseline results on both QASENT and WIKIQA datasets. Questions without correct answers in the candidate sentences are removed in the WIKIQA dataset. The best results are in **bold**.

Model	Prec	Rec	F_1
CNN-Cnt	26.09	37.04	30.61
+QLen	27.96	37.86	32.17
+SLen	26.14	37.86	30.92
+QClass	27.84	33.33	30.34
+All	28.34	35.80	31.64

Table 5: Evaluation of answer triggering on the WIKIQA dataset. Question-level precision, recall and F_1 scores are reported.

makes use of rich lexical semantic features, including word/lemma matching, WordNet and vector-space lexical semantic models. We do not include features for Named Entity matching.[5]

We include two sentence semantic methods, Paragraph Vector[6] (PV; Le and Mikolov, 2014) and Convolutional Neural Networks (CNN; Yu et al., 2014). The model score of PV is the cosine similarity score between the question vector and the sentence vector. We follow Yu et al. (2014) and employ a bigram CNN model with average pooling. We use the pre-trained word2vec embeddings provided by Mikolov et al. (2013) as model input.[7] For computational reasons, we truncate sentences up to 40 tokens for our CNN models.

Finally, we combine each of the two sentence semantic models with the two word matching features by training a logistic regression classifier, referring as PV-Cnt and CNN-Cnt. CNN-Cnt has been shown to achieve state-of-the-art results on the QASENT dataset (Yu et al., 2014).

3.2 Evaluation of Answer Triggering

The task of answer sentence selection assumes that there exists at least one correct answer in the candidate answer sentence set. Although the assumption simplifies the problem of question answering, it is unrealistic for practical QA systems. Modern QA systems rely on an independent component to pre-select candidate answer sentences, which utilizes various signals such as lexical matching and user behaviors. However, the candidate sentences are not guaranteed to contain the correct answers, no matter what kinds of pre-selection components are employed. We propose the answer triggering task, a new challenge for the question answering problem, which requires QA systems to: (1) detect whether there is at least one correct answer in the set of candidate sentences for the question; (2) if yes, select one of the correct answer sentences from the candidate sentence set.

Previous work adopts MAP and MRR to evaluate the performance of a QA system on answer sentence selection. Both metrics evaluate the relative ranks of correct answers in the candidate sentences of a question, and hence are not suitable for evaluating the task of answer triggering. We need metrics that consider both the presence of answers with respect to a question and the correctness of system predictions.

We employ precision, recall and F_1 scores for answer triggering, at the question level. In particular, we compute these metrics by aggregating all the candidate sentences of a question. A question is treated as a positive case only if it contains one or more correct answer sentences in its candidate sentence pool. For the prediction of a question, we only consider the sentence in the candidate set that has the highest model score. If the score is above a predefined threshold and the sentence is labeled as a correct answer to the question, then it means that the prediction is correct and the question is answered correctly.

3.3 Results

WIKIQA vs. QASENT The MAP and MRR results are presented in Table 4. We only evaluate questions with answers in the WIKIQA dataset under these metrics. On the QASENT dataset, as found by prior work, the two word matching methods are very strong baselines, in which they sig-

[5]The improvement gains from the features are marginal on the QASENT dataset.

[6]We choose the Distributed Bag of Words version of Paragraph Vector, as we found it significantly outperforms the Distributed Memory version of Paragraph Vector.

[7]Available at https://code.google.com/p/word2vec/

nificantly outperform sentence semantic models. By incorporating rich lexical semantic information, LCLR further improves the results. CNN-Cnt gives results that match LCLR, and PV-Cnt performs worse than CNN-Cnt.[8]

The story on the WIKIQA dataset is different. First, methods purely rely on word matching are not sufficient to achieve good results. Second, CNN significantly outperforms simple word matching methods and performs slightly better than LCLR, which suggests that semantic understanding beyond lexical semantics is important for obtaining good performance on WIKIQA. Finally, word matching features help to further boost CNN results by approximately 3 to 4 points in both MAP and MRR.

Evaluation of answer triggering on WIKIQA We evaluate the best system CNN-Cnt on the task of answer triggering, and the results are shown in Table 5. We tune the model scores for making predictions with respect to F_1 scores on the dev set, due to the highly skewed class distribution in training data. The absolute F_1 scores are relative low, which suggests a large room for improvement.

We further study three additional features: the length of question (QLen), the length of sentence (SLen), and the class of the question (QClass). The motivation for adding these features is to capture the hardness of the question and comprehensiveness of the sentence. Note that the two question features have no effects on MAP and MRR. As shown in Table 5, the question-level F_1 score is substantially improved by adding a simple QLen feature. This suggests that designing features to capture question information is very important for this task, which has been ignored in the past. SLen features also give a small improvement in the performance, and QClass feature has slightly negative influence on the results.

3.4 Error Analysis & Discussion

The experimental results show that for the same model, the performance on the WIKIQA dataset is inferior to that on the QASENT dataset, which suggests that WIKIQA is a more challenging dataset. Examining the output of CNN-Cnt, the best performing model, on the WIKIQA dev set seems to suggest that deeper semantic understanding and answer inference are often required. Below are two examples selected that CNN-Cnt does not correctly rank as the top answers:

Q1: *What was the GE building in Rockefeller Plaza called before?*
A1: [GE Building] *Known as the RCA Building until 1988, it is most famous for housing the headquarters of the television network NBC.*

Q2: *How long was I Love Lucy on the air?*
A2: [I Love Lucy] *The black-and-white series originally ran from October 15, 1951, to May 6, 1957, on the Columbia Broadcasting System (CBS).*

Answering the first question may require a better semantic representation that captures the relationship between "called before" and "known ... until". As for the second question, knowing that on a TV channel (e.g., CBS) implies "on the air" and a time span between two dates is legitimate to a "how long" question is clearly beneficial.

4 Conclusion

We present WIKIQA, a new dataset for open-domain question answering. The dataset is constructed in a natural and realistic manner, on which we observed different behaviors of various methods compared with prior work. We hope that WIKIQA enables research in the important problem of answer triggering and enables further research in answer sentence selection in more realistic settings. We also hope that our empirical results will provide useful baselines in these efforts.

References

Michael Heilman and Noah A. Smith. 2010. Tree edit models for recognizing textual entailments, paraphrases, and answers to questions. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 1011–1019.

Quoc Le and Tomas Mikolov. 2014. Distributed representations of sentences and documents. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 1188–1196.

Xin Li and Dan Roth. 2002. Learning question classifiers. In *Proceedings the International Conference on Computational Linguistics (COLING)*, pages 556–562.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S. Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Proceedings of the Conference on Advances*

[8] Our CNN reimplementation performs slightly worse than (Yu et al., 2014).

in Neural Information Processing Systems (NIPS), pages 3111–3119.

Aliaksei Severyn and Alessandro Moschitti. 2013. Automatic feature engineering for answer selection and extraction. In *Proceedings of the Conference on Empirical Methods for Natural Language Processing (EMNLP)*, pages 458–467.

Ellen M. Voorhees and Dawn M. Tice. 1999. The TREC-8 question answering track evaluation. In *Proceedings of the Text Retrieval Conference TREC-8*, page 82.

Mengqiu Wang, Noah A Smith, and Teruko Mitamura. 2007. What is the Jeopardy model? A quasi-synchronous grammar for QA. In *Proceedings of the Conference on Empirical Methods for Natural Language Processing (EMNLP)*, pages 22–32.

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 Annual Meeting of the North American Association of Computational Linguistics (NAACL)*, pages 858–867.

Wen-tau Yih, Ming-Wei Chang, Christopher Meek, and Andrzej Pastusiak. 2013. Question answering using enhanced lexical semantic models. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*.

Lei Yu, Karl Moritz Hermann, Phil Blunsom, and Stephen Pulman. 2014. Deep learning for answer sentence selection. In *Proceedings of the Deep Learning and Representation Learning Workshop: NIPS-2014*.