Beyond Factoid QA: Effective Methods for Non-factoid Answer Sentence Retrieval

Liu Yang[1], Qingyao Ai[1], Damiano Spina[2], Ruey-Cheng Chen[2], Liang Pang[3],
W. Bruce Croft[1], Jiafeng Guo[3], and Falk Scholer[2]

[1] Center for Intelligent Information Retrieval, University of Massachusetts Amherst, MA, USA
[2] RMIT University, Melbourne, Australia
[3] Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China
{lyang, aiqy, croft}@cs.umass.edu, {damiano.spina, ruey-cheng.chen, falk.scholer}@rmit.edu.au, guojiafeng@ict.ac.cn, pangliang@software.ict.ac.cn

Abstract. Retrieving finer grained text units such as passages or sentences as answers for non-factoid Web queries is becoming increasingly important for applications such as mobile Web search. In this work, we introduce the answer sentence retrieval task for non-factoid Web queries, and investigate how this task can be effectively solved under a learning to rank framework. We design two types of features, namely semantic and context features, beyond traditional text matching features. We compare learning to rank methods with multiple baseline methods including query likelihood and the state-of-the-art convolutional neural network based method, using an answer-annotated version of the TREC GOV2 collection. Results show that features used previously to retrieve topical sentences and factoid answer sentences are not sufficient for retrieving answer sentences for non-factoid queries, but with semantic and context features, we can significantly outperform the baseline methods.

1 Introduction

A central topic in developing intelligent search systems is to provide answers in finer-grained text units, rather than to simply rank lists of documents in response to Web queries. This can not only save the users' efforts in fulfilling their information needs, but also will improve the user experience in applications where the output bandwidth is limited, such as mobile Web search and spoken search. Significant progress has been made at answering factoid queries [18, 22], such as "how many people live in Australia?", as defined in the TREC QA track. However, there are diverse Web queries which cannot be answered by a short fact, ranging from advice on fixing a mobile phone, to requests for opinions on some public issues. Retrieving answers for these "non-factoid" queries from Web documents remains a critical challenge in Web question answering (WebQA).

Longer answers are usually expected for non-factoid Web queries, such as sentences or passages. However, research on passage-level answer retrieval can be difficult due to both the vague definition of a passage and evaluation methods [10]. A more natural and direct approach is to focus on retrieving sentences that are part of answers. Sentences are basic expression units in most if not all natural languages, and are easier to define and evaluate compared with passages. Therefore, in this paper, we introduce answer sentence retrieval for non-factoid Web queries as a practical WebQA task. To facilitate research on this task, we have created a benchmark data set referred as *WebAP* using a Web collection (TREC GOV2). To investigate the problem, we propose the first research question:

RQ1 Could we directly apply existing methods like factoid QA methods and sentence selection methods to solve this task?

Methods on factoid QA and sentence retrieval/selection are closely related work for our task. Factoid QA has been studied for some time, facilitated by the TREC QA track, and a state-of-the-art method is to use a convolutional neural network (CNN) [22] based model. In our study, we first adopt the CNN based method for factoid QA [22] and a sentence selection method using machine learning technology [13] for our task. However, we obtain inferior results for these advanced models, as compared with traditional retrieval models (i.e., language model). The results indicate that automatically learned word features (as in CNN) and simple text matching features (as in the sentence selection method) may not be sufficient for the answer sentence retrieval task. This leads to our second research question:

RQ2 How could we design more effective methods for answer sentence retrieval for non-factoid Web queries?

Previous results show that retrieving sentences that are part of answers is a more challenging task, requiring more powerful features than traditional relevance retrieval. By analyzing the task, we make two key observations from the WebAP data:

1. Due to the shorter length of sentences compared with documents, the problem of vocabulary-mismatch may be even more severe in answer sentence retrieval for non-factoid Web queries. Thus in addition to text matching features, we need more features to capture the semantic relations between query and answer sentences.
2. Non-factoid questions usually require multiple sentences as answers, and these answer sentences do not scatter in documents but often form small clusters. Thus the context of a sentence may be an important clue for identifying answer sentences.

Based on these observations, we design and extract two new types of features for answer sentence retrieval, namely *semantic features* and *context features*. We adopt learning to rank (L2R) models for sentence ranking, which have been successfully applied to document retrieval, collaborative filtering and many other applications. The experimental results show that significant improvement over existing methods can be achieved by our ranking model with semantic and context features.

The contributions of this paper can be summarized as follows:

1. We formally introduce the answer sentence retrieval task for non-factoid Web queries, and build a benchmark data set (WebAP) using the TREC GOV2 collection. We show that a state-of-the-art method from research on TREC QA track data does not work for this task, indicating answer sentence retrieval for non-factoid Web queries is a challenging task that requires the development of new methods. We released this data set to the research community.
2. Based on the analysis of the WebAP data, we design effective new features including semantic and context features for non-factoid answer sentence retrieval. We analyze the performance of different feature combinations to show the relative feature importance.

3. We perform a thorough experimental study with sentence level answer annotations. The results show that MART with semantic and context features can significantly outperform existing methods including language models, a state-of-the-art CNN based factoid QA method and a sentence selection method using multiple features.

The rest of the paper is organized as follows. The next section, §2, is an introduction to related work. We describe task definition and data for non-factoid answer sentence retrieval in §3. §4 is a baseline experiment where we attempt to apply existing standard techniques to confirm whether a new set of techniques is needed for our task. §5 describes the proposed semantic features and context features. §6 is a systematic experimental analysis using the annotated TREC GOV2 collection. §7 gives the conclusions and discusses future work.

2 Related Work

Our work is related to several research areas, including answer passage retrieval, answer retrieval with translation models, answer ranking in community question answering (CQA) sites and answer retrieval for factoid questions.

Answer Passage Retrieval. Keikha et al. [9, 10] developed an annotated data set for non-factoid answer finding using TREC GOV2 collections and topics. They annotated passage-level answers, revisited several passage retrieval models with this data, and came to the conclusion that the current methods are not effective for this task. Our research work departs from Keikha et al. [9, 10] by developing methods for answer sentence retrieval.

Answer Retrieval with Translation Models. Some previous research on answer retrieval has been based on statistic translation models to find semantically similar answers [1, 15, 20]. Xue et al. [20] proposed a retrieval model that combines a translation-based language model for the question part with a query likelihood approach for the answer part. Riezler et. al. [15] presented an approach to query expansion in answer retrieval that uses machine translation techniques to bridge the lexical gap between questions and answers. Berger et al. [1] studied multiple statistical methods such as query expansion, statistical translation, and latent variable models for answer finding.

Answer Ranking in CQA. Surdeanu et al. [16] investigated a wide range of feature types including similarity features, translation features, density / frequency features, Web correlation features for ranking answers to non-factoid questions in Yahoo! Answers. Jansen et al. [8] presented an answer re-ranking model for non-factoid questions that integrate lexical semantics with discourse information driven by two representations of discourse. Answer ranking in CQA sites is a somewhat different task than answer retrieval for non-factoid questions: answer sentences could come from multiple documents for general non-factoid question answering, and the candidate ranked answer set is much smaller for a typical question in CQA sites.

Answer Retrieval for Factoid Questions. There has also been substantial research on answer sentence selection with data from TREC QA track [17, 18, 21, 22]. Yu et al. [22] proposed an approach to solve this task via means of distributed representations, and learn to match questions with answers by considering their semantic encoding. Although the target answers are also at the sentence level, this research is focused on factoid questions. Our task is different in that we investigate answer sentence retrieval for non-factoid questions. We also compare our proposed methods with a state-of-the-art factoid QA method to show the advantages of developing techniques specifically for non-factoid answer data.

3 Task Definition and Data

We now give a formal definition of our task. Given a set of non-factoid questions $\{Q_1, Q_2, \cdots, Q_n\}$ and Web documents $\{D_1, D_2, ...D_m\}$ that may contain answers, our task is to learn a ranking model **R** to rank the sentences in the Web documents to find sentences that are part of answers. The ranker is trained based on available features F_S and labels L_S to optimize a metric **E** over the sentence rank list.

Our task is different from previous research in the TREC QA track and answer retrieval in CQA sites like Yahoo! Answers. Unlike the TREC QA track, we are focusing on answer finding for non-factoid questions. Unlike the research on answer retrieval in CQA, we aim to find answers from general Web documents, not limited to CQA answer posts. As a consequence, this task is more challenging for two reasons: answers could be much longer than in factoid QA, and the search space is much larger than CQA answer posts.

A test collection of questions and multi-sentence (passage) answers has been created based on the TREC GOV2 queries and documents.[1] GOV2 is the test collection used for the TREC Terabyte Track and crawled from .gov sites in early 2004. It contains 25 million documents. The annotated GOV2 data set was produced by the following process [9, 10]. For each TREC topic that was likely to have passage-level answers (82 in total), the top 50 documents were retrieved using a state-of-the-art retrieval model [7, 11]. From the retrieved documents, documents identified as relevant in the TREC judgments were annotated for answer passages. Passages were marked with labels "Perfect", "Excellent", "Good", "Fair". The annotators found 8027 answer passages to 82 TREC queries, which is 97 passages per query on average. Among the annotated passages, which exclude passages without annotations and are treated as negative instances, 43% are perfect answers, 44% are excellent, 10% are good and the rest are fair answers.

To obtain the annotation for our answer sentence retrieval task, we let sentences in answer passages inherit the label of the passage. Then we map "Perfect", "Excellent", "Good", "Fair" to $4 \sim 1$ and assign 0 for all the other sentences. Note that there are some duplicate sentences in the previously annotated passage data set. Judgments over these duplicates are not entirely consistent. We fix this problem by a majority vote and break ties by favoring more relevant labels. The data after label inheritance and inconsistent judgment fixing is named *WebAP* data. There are 991233 sentences in the data set and the average length of sentences is 17.58. After label propagation from passage level to sentence level, 99.02% (981510) sentences are labeled as 0 and less than 1% sentences have positive labels (149 sentences are labeled as 1; 783 sentences are labeled as 2; 4283 sentences are labeled as 3; 4508 sentences are labeled as 4.) . Highly imbalanced labels make this task even more difficult.

To better understand our task and data, we show a comparison of sample questions and answers in TREC QA Track data and WebAP data in Table 1. We can clearly see the difference is that answers in TREC QA Track data are mostly short phrases like entities and numbers, whereas answers in WebAP data are longer sentences.

[1] The data set is publicly available at https://ciir.cs.umass.edu/downloads/WebAP/.

Table 1: Comparison of sample questions and answers in TREC QA Track data and WebAP data.

Sample Questions and Answers in TREC QA Track Data
Query 201: Question: What was the name of the first Russian astronaut to do a spacewalk? Answer: Aleksei A. Leonov Answer Document ID: LA072490-0034
Query 202: Question: Where is Belize located? Answer: Central America Answer Document ID: FT934-14974
Query 203: Question: How much folic acid should an expectant mother get daily? Answer: 400 micrograms Answer Document ID: LA061490-0026

Sample Questions and Answer Sentences with Labels in WebAP Data
Query 704: Question: What are the goals and political views of the Green Party? Answer: (*Perfect*) The Green Party promotes an ecological vision which understands that all life on our planet is interconnected; that cooperation is more essential to our well-being than competition; and that all people are connected to and dependent upon one another and upon the natural systems of our world. (*Excellent*) The need for fairness for people and local communities in developing economic opportunities, instead of continually favoring big corporations and other concentrations of wealth and power; and a fair, equitable, progressive tax system. (*None*) Statements were supplied by the political parties and have not been checked for accuracy by any official agency.
Query 847: Question: What was the role of Portugal in World War II? Answer: (*Perfect*) The Allies began this economic war with some advantages, most notably a centuries-old Anglo-Portuguese alliance, coupled with close economic and trade relations (Britain was Portugal's leading trade partner), as well as Portugal's dependence on U.S. petroleum, coal, and chemical supplies. (*Good*) Portugal, Spain, Sweden, and Turkey all sold Nazi Germany resources critical to sustaining its wartime industry. (*Fair*) During WWII, Switzerland served as a curtain for other countries by creating multinational gold depository for neutral and non-aligned national states — Sweden, Portugal, Spain and Turkey — to use in trading money with the Axis. (*None*) The records of claims reflect attempts at achieving diplomatic remedies to losses suffered in federal courts in suits handled by the Office of Alien Property.

4 Baseline Experiments

Our first task on the WebAP data set is to seek solution of RQ1, in which we ask if a new set of techniques is needed. We address this question using a baseline experiment, in which we use some techniques that should be reasonable for retrieving non-factoid answers, and compare these techniques with a factoid question answering method.

We set up this experiment by including the following three classes of techniques:

1. **Retrieval functions.** In this experiment, we considered query likelihood language model with Dirichlet smoothing (LM).
2. **Factoid question answering method.** In this experiment, we use a more recent approach based on convolutional neural network (CNN) [22], whose performance is current on par with the best results on TREC QA Track data. This is a supervised method and needs to be trained with pre-defined word embeddings. Two variants are tested here, which are CNN with word count features and CNN without word count features.
3. **Summary sentence selection method.** In this experiment, we test a L2R approach proposed by Metzler and Kanungo [13], which uses 6 simple features referred as *MK features* as described in §5 to address the lexical matching between the query and

Table 2: Baseline experiment results for non-factoid answer sentence retrieval of different methods. The best performance is highlighted in boldface. ‡ means significant difference over CNN with word count features with $p < 0.01$ measured by the Student's paired t-test.

Feature Set	**Model**	**NDCG@10**	**P@10**	**MRR**
CNN(No word count)	CNN	0.0218	0.0341	0.0909
CNN(With word count)		0.0596	0.0646	0.1254
MK	MART	0.1163 ‡	0.1293 ‡	0.2677 ‡
LM	Base	**0.1340** ‡	**0.1451** ‡	**0.3395** ‡

sentences. As suggested in the original paper, we use the MART ranking algorithm to combine these features.

The results are given in Table 2. LM and MK perform better than CNN based methods. LM achieves the best results under all the three metrics. CNN based methods perform poorly on this task. Using word count features in [22], CNN gets slightly better results. However, only using LM can achieve 170.73% gain for MRR over CNN with word count features and the difference is statistically significant measured by the Student's paired t-test ($p < 0.01$). MK achieves better performance than CNN based methods, but it performs worse than LM. This result shows that automatically learned word features (as in CNN) and simple combined text matching features (as in MK) may not be sufficient for our task, suggesting that a new set of techniques is needed for non-factoid answer sentence retrieval.

5 Beyond Factoid QA: Capturing Semantics and Context

In the previous experiment, we found that a state-of-the-art factoid question answering method is not very effective on our task. L2R with MK features and LM performs better than CNN based methods. We now construct a ranker using MK features which include LM. We further add semantic and context features for answer sentence retrieval of non-factoid questions. First, we give a detailed description of the MK features as follows.

ExactMatch. Exact match is a binary feature indicating whether the query is a substring of the sentence.

TermOverlap. Term overlap measures the number of terms that are both in the query and the sentence after stopping and stemming.

SynonymsOverlap. Synonyms overlap is the fraction of query terms that have a synonym (including the original term) in the sentence, computed by using Wordnet.[2]

LanguageModelScore. Language model score is computed as the log likelihood of the query being generated from the sentence. The sentence language model is smoothed using Dirichlet smoothing. This feature is essentially the query likelihood language model score[3]. The feature is computed as:

$$f_{LM}(Q,S) = \sum_{w \in Q} tf_{w,Q} log \frac{tf_{w,S} + \mu P(w|C)}{|S| + \mu} \qquad (1)$$

where $tf_{w,Q}$ is the number of times that w occurs in the query, $tf_{w,S}$ is the number of times that w occurs in the sentence, $|S|$ is the length of sentences, $P(w|C)$ is the background

[2] http://wordnet.princeton.edu/

language model and μ is a parameter for Dirichlet smoothing. Note that the parameter μ tends to be smaller than the case in document retrieval because the average length of sentences is much shorter than that of documents.

SentLength. Sentence length is the number of terms in the sentence after stopping.

SentLocation. Sentence location is the relative location of the sentence within the document, computed as the position divided by the total number of sentences.

5.1 Semantic Features

Short text units such as sentences are more likely to suffer from query mismatch. The same topics could get radically different wordings between the questions and answers, which leads to the "lexical chasm" problem [1]. Thus we consider the following semantic features to handle this problem.

ESA. Explicit Semantic Analysis (ESA) [6] is a method that represents text as a weighted mixture of a predetermined set of natural concepts defined by Wikipedia articles which can be easily explained. Semantic relatedness is computed as the cosine similarity between the query ESA vector and the sentence ESA vector. A recent dump of English Wikipedia (June 2015) is used to generate ESA representations for queries and sentences.

WordEmbedding. Word embeddings are continuous vector representations of words learned from large amount of text data using neural networks. Words with similar meanings are also in close distances in this vector space. Mikolov et al. [14] introduced the Skip-gram model as an efficient method for learning high quality vector representations of words from large amounts of unstructured text data. The implementation of the continuous bag-of-words and skip-gram architectures for computing vector representations of words is released as an open-source project Word2Vec.[3] We compute this feature as the average pairwise cosine similarity between any query-word vector and any sentence-word vector following previous work [2].

EntityLinking. Linking short texts to a knowledge base to obtain the most related concepts gives an informative semantic representation that can be used to represent queries and sentences. We generate such a representation through an entity linking system Tagme [4], which is able to efficiently augment a plain text with pertinent hyperlinks to Wikipedia pages. On top of this, we produce the following semantic feature: the Jaccard similarity between the set of Wikipedia pages linked by Tagme to the query q and the set of pages linked to a given sentence s (Eq. 2):

$$\text{TagmeOverlap}(q, s) = \frac{\text{Tagme}(q) \cap \text{Tagme}(s)}{\text{Tagme}(q) \cup \text{Tagme}(s)}. \tag{2}$$

ESA takes advantages of human labeled concepts from Wikipedia and explicitly presents topical differences between queries and sentences; **WordEmbedding** captures word semantics from unstructured text data based on the hypothesis that words with similar meanings should appear in similar lingual contexts; and **EntityLinking** converts sentences to entity space where different words for the same entity, and different entities with similar Wikipedia pages are assigned with high similarity scores. Combined together, these semantic features compensate MK features and alleviate problems like query mismatching through the introduction of a semantic similarity score.

[3] https://code.google.com/p/word2vec/

5.2 Context Features

Context features are features specific to the context of the candidate sentence. We define the context of a sentence as the adjacent sentence before and after it. The intuition is that the answer sentences are very likely to be surrounded by other answer sentences. Context features could be generated based on any sentence features. They include features in the following two types:

- **<Feature>SentenceBefore:** MK features and semantic features of the sentence before the candidate sentence.
- **<Feature>SentenceAfter:** MK features and semantic features of the sentence after the candidate sentence.

So applying this idea to a set of n features, we produce $2n$ new features. In our experiments, we define $context()$ as a procedure which could be applied to MK features and semantic features. So there are 12 context features for MK features and 6 context features for semantic features.

5.3 Learning Models

With the computed features, we carry out sentence re-ranking with L2R methods using the following models:

- **MART** Multiple Additive Regression Trees [5], also known as gradient boosted regression trees, produces a prediction model in the form of an ensemble of weak prediction models.
- **LambdaMART** [19] combines the strengths of MART and LambdaRank which has been shown to be empirically optimal for a widely used information retrieval measure. It uses the LambdaRank gradients when training each tree, which could deal with highly non-smooth IR metrics such as DCG and NDCG.
- **Coordinate Ascent (CA)** [12] is a list-wise linear feature-based model for information retrieval which uses coordinate ascent to optimize the model's parameters. It optimizes the objective by iteratively updating each dimension while holding other dimensions fixed.

6 Experiments

6.1 Experimental Settings

As presented in §3, we use the aforementioned WebAP dataset as the benchmark. We follow the LETOR experiment protocol, partitioning data by queries and conducting 5-fold cross validation throughout. We use our own package *SummaryRank*[4] for feature extraction. We use RankLib[5] to implement MART and CA, and use *jforests*[6] to implement LambdaMART. For each experimental run, we optimize the hyperparameters using

[4] http://rmit-ir.github.io/SummaryRank

[5] http://www.lemurproject.org/ranklib.php

[6] https://code.google.com/p/jforests/

cross validation and report the best performance. [7] For evaluation, we compute a variety of metrics including NDCG@10, P@10 and MRR, focusing on the accuracy of top ranked answer sentences in the results.

6.2 Overall Analysis of Results

Table 3 shows the evaluation results for non-factoid answer sentence retrieval using different feature sets and learning models. We summarize our observations as follows: (1) For feature set comparison, the results are quite consistent across the three different learning models where the combination of MK, semantic features, and context features achieve the best results for all three learning model settings. For MRR, this combination achieves 5.4% gain using CA, 68.55% gain using MART, and 10.05% gain using LambdaMART over the MK feature set. Similar gains are also observed under other metrics. In terms of relative feature importance, context features achieve larger gain compared to semantic features although adding both of them can improve the performance of the basic MK feature set. (2) For the learning model comparison, MART achieves the best performance with MK + Semantics + Context(All) features, with statistically significant differences with respect to both LM and MK(MART), which shows the effectiveness of MART for combining different features to learn a ranking model for non-factoid answer sentence retrieval.

6.3 Effect of Semantic and Context Features

As mentioned previously, retrieval on short text such as sentences are vulnerable to problems such as "lexical chasm" [1]. By introducing semantic features, we attempt to address this problem through matching query and sentences in semantic space. The experiments indicate that semantic features indeed benefit retrieval effectiveness. As shown in Table 3, MK + Semantics with MART obtains 3.78% gain in NDCG@10, 3.71% in P@10, 1.94% in MRR. Similar improvements can also be found in results with other L2R models including LambdaMART and CA except that we observe slightly loss under NDCG@10 and MRR for CA. Overall, the three simple but effective semantic features provide important information for non-factoid answer sentence retrieval.

Our intuition for the design of context features is that good answer sentences are likely to be surrounded by other answer sentences. Our experimental results also demonstrate the effectiveness of context features. In Table 3, we observe considerable improvement with context features: MK + Semantics + Context(All) outperforms MK + Semantics in NDCG@10, P@10 and MRR for 13.01%, 11.18%, 17.19% with LambdaMART, 54.43%, 50.93%, 65.34% with MART and 20.79%, 14.63%, 7.64% with CA. Thus the performance of these three learning models could be improved by incorporating context features.

[7] We just report the hyperparameters used for MK + Semantics + Context(All) feature sets given the space limit. The hyperparameters for other feature sets can be obtained by standard 5-fold cross validation. For parameter values in MART, we set the number of trees as 100, the number of leaves of each tree as 20, learning rate as 0.05, the number of threshold candidates for tree splitting as 256, min leaf support as 1, the early stop parameter as 100. For parameter values in CA, we set the number of random restarts as 3, the number of iterations to search in each dimension as 25, performance tolerance between two solutions as 0.001 . For parameter values in LambdaMART, we set the number of trees as 1000, the number of leaves of each tree as 15, learning rate as 0.1, minimum instance percentage per leaf as 0.25, feature sampling ratio as 0.5. We empirically set the parameter $\mu = 10$ in the computation of the *LanguageModelScore* feature.

Table 3: Evaluation results for non-factoid answer sentence retrieval of different feature sets and learning models. “Context(All)” denotes context features for both MK and semantics features. The best performance is highlighted in boldface. † and ‡means significant difference over LM with $p < 0.05$ and $p < 0.01$ respectively measured by the Student’s paired t-test. ** means significant difference over MK with the same learning model with $p < 0.01$ measured by the Student’s paired t-test.

Feature Set	**Model**	**NDCG@10**	**P@10**	**MRR**
LM	Base	0.1340	0.1451	0.3395
MK	Coord. Ascent	0.1590	0.1524	0.3860
MK + Semantics		0.1486	0.1585	0.3781
MK + Semantics + Context(All)		**0.1795** ‡	**0.1817** †	**0.4070**
MK	MART	0.1163	0.1293	0.2677
MK + Semantics		0.1207	0.1341	0.2729
MK + Semantics + Context(All)		**0.1864** $^{\dagger **}$	**0.2024** $^{\dagger **}$	**0.4512** $^{\dagger **}$
MK	LambdaMART	0.1441	0.1537	0.3662
MK + Semantics		0.1591	0.1744	0.3439
MK + Semantics + Context(All)		**0.1798** †	**0.1939** †	**0.4030**

6.4 Examples of Top Ranked Sentences

We further show some examples and analysis of top-1 ranked sentences by different methods in Table 4. In general, the top-1 ranked sentences by MK + Semantics + Context(All) are better than the other two methods. Although there are lexical matches for top-1 sentences retrieved by all methods, sentences with lexical matches may be not answer sentences of high quality. For example, for query 808, MK features will be confused on “Korean”, “North Korean” and “South Korean” appearing frequently in sentences since there is a common term among them. Semantic features play an important role here to alleviate this problem. Semantic features such as **EntityLinking** can differentiate “Korean” with “North Korean” by linking them to different Wikipedia pages.

Context features have the potential to guide the ranker in the right direction since correct non-factoid answer sentences are usually surrounded by other similar non-factoid answer or relevant sentences. Query 816 in Table 4 gives one example for their effects. Without context features, LambdaMART with MK+Semantics features retrieved a non-relevant sentence as the top-1 result. The retrieved sentence may seem relevant itself (as it indeed mentions USAID’s efforts to support biodiversity), but if we pull out its context:

- SentenceBefore (None): *... BIOFOR embarked upon a global study of lessons learned from USAID’s community-based forest management projects since 1985.*
- SentenceRetrieved (None): *USAID forestry programs support biodiversity efforts by helping to protect habitats for forest inhabitants.*
- SentenceAfter (None): *The African continent contains approximately 650 million hectares of forests ...*

The sentences around the *SentenceRetrieved* are not relevant to the query and are labeled as “None”. From its context, we can see that the *SentenceRetrieved* is actually not talking about Galapagos Islands. Comparing to that, the context of top-1 sentence retrieved by LambdaMART with MK + Semantics + Context(All) features are:

- SentenceBefore (None): *Relations between ... in the Galapagos have greatly improved ... and approval of the Galapagos Marine Reserve Management Plan.*

Table 4: Examples of top-1 ranked sentences by using different methods. Integers in the 2nd column are relevance levels (0 = not relevant). Lexically matched sentence terms are underlined.

770: *What is the state of Kyrgyzstan United States relations?*

MK	0	They seized hostages and several villages, allegedly seeking to create an Islamic state in south Kyrgyzstan as a springboard for a jihad in Uzbekistan.
MK+Sem	3	In 1994, Kyrgyzstan concluded a bilateral investment treaty with the United States, and in 1999 Kyrgyzstan became a member of the WTO.
MK+Sem+Context	4	The extension of unconditional normal trade relations treatment to the products of Kyrgyzstan will permit the United States to avail itself of all rights under the WTO with respect to Kyrgyzstan.

808: *What information is available on the involvement of the North Korean Government in counterfeiting of US currency?*

MK	0	U.S. firms report that, although the release of confidential business information is forbidden by Korean law, government officials have not sufficiently protected submitted information and in some cases, proprietary information has been made available to Korean competitors or to their trade associations.
MK+Sem	3	North Korean regime continues to export weapons and engage in state sponsored international crime to include narcotics trafficking, and counterfeiting U.S. currency.
MK+Sem+Context	3	While the State Department reports that North Korea's most profitable operation involves counterfeiting U.S. currency,(101) government-controlled operations also counterfeit a range of other countries' currencies and other trademarked products.

816: *Describe efforts made by USAID to protect the biodiversity in the Galapagos Islands in Ecuador.*

MK	0	efforts in the Galapagos Islands and other key areas of Ecuador, which has the greatest biodiversity per hectare of any country in South America.
MK+Sem	0	USAID forestry programs support biodiversity efforts by helping to protect habitats for forest inhabitants.
MK+Sem+Context	3	In addition, USAID contributed to the establishment of a capital fund for the benefit of the Charles Darwin Foundation in the Galapagos Islands to guide the development of the sustainable, scientifically based, participatory management of the Galapagos ecosystem.

– SentenceRetrieved (Perfect): ... *USAID contributed to ... in the Galapagos Islands to guide the development of the sustainable, ... management of the Galapagos ecosystem.*

– SentenceAfter (Excellent): *USAID is also assisting ... for sustainable forestry development and biodiversity conservation in Ecuador.*

SentenceBefore and *SentenceAfter* are talking about Galapagos Islands and USAID's work for biodiversity conservation in Ecuador, and *SentenceAfter* is labeled as an "Excellent" answer sentence. With context features, *SentenceRetrieved* is promoted as the top-1 result due to its good context, and our empirical experiments show that these promotions are beneficial for model effectiveness (the promoted sentence in this case is indeed a perfect answer for query 816).

7 Conclusions and Future Work

In this paper, we formally introduced the answer sentence retrieval task for non-factoid Web queries and investigated a framework based on learning to rank methods. We compared learning to rank methods with baseline methods including language models and a CNN based method. We found that both semantic and context features are useful for non-factoid answer sentences retrieval. In particular, the results show that MART with appropriate features outperforms all the baseline methods significantly under multiple metrics and provides a good basis for non-factoid answer sentence retrieval.

For future work, we would like to investigate more features such as syntactic features and readability features to further improve non-factoid answer sentence retrieval. Learning an effective representation of answer sentences for information retrieval is also an interesting direction to explore.

Acknowledgments. This work was partially supported by the Center for Intelligent Information Retrieval, by NSF grant #IIS-1160894, by NSF grant #IIS-1419693, by ARC Discovery grant DP140102655 and by ARC Project LP130100563. Any opinions, findings and conclusions expressed in this material are those of the authors and do not necessarily reflect those of the sponsor. We thank Mark Sanderson for the valuable comments on this work.

References

[1] A. Berger, R. Caruana, D. Cohn, D. Freitag, and V. Mittal. Bridging the Lexical Chasm: Statistical Approaches to Answer-finding. In *Proceedings of SIGIR'00*, 2000.

[2] R.-C. Chen, D. Spina, W. B. Croft, M. Sanderson, and F. Scholer. Harnessing semantics for answer sentence retrieval. In *Proceedings of ESAIR'15*, 2015.

[3] W. B. Croft, D. Metzler, and T. Strohman. *Search Engines: Information Retrieval in Practice*. Addison-Wesley Publishing Company, USA, 1st edition, 2009.

[4] P. Ferragina and U. Scaiella. Fast and accurate annotation of short texts with wikipedia pages. *Software, IEEE*, 29(1):70–75, 2012.

[5] J. H. Friedman. Greedy Function Approximation: A Gradient Boosting Machine. *Annals of Statistics*, 29:1189–1232, 2000.

[6] E. Gabrilovich and S. Markovitch. Computing Semantic Relatedness Using Wikipedia-based Explicit Semantic Analysis. In *Proceedings of IJCAI'07*, 2007.

[7] S. Huston and W. B. Croft. A comparison of retrieval models using term dependencies. In *Proceedings of CIKM'14*, 2014.

[8] P. Jansen, M. Surdeanu, and P. Clark. Discourse Complements Lexical Semantics for Non-factoid Answer Reranking. In *Proceedings of ACL'14*, pages 977–986.

[9] M. Keikha, J. H. Park, and W. B. Croft. Evaluating Answer Passages Using Summarization Measures. In *Proceedings of SIGIR'14*, 2014.

[10] M. Keikha, J. H. Park, W. B. Croft, and M. Sanderson. Retrieving Passages and Finding Answers. In *Proceedings of ADCS'14*, pages 81–84, 2014.

[11] D. Metzler and W. B. Croft. A Markov Random Field Model for Term Dependencies. In *Proceedings of SIGIR'05*, 2005.

[12] D. Metzler and W. B. Croft. Linear Feature-based Models for Information Retrieval. *Inf. Retr.*, 10(3):257–274, June 2007.

[13] D. Metzler and T. Kanungo. Machine Learned Sentence Selection Strategies for Query-Biased Summarization. In *Proceedings of SIGIR 2008 Learning to Rank Workshop*.

[14] T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient estimation of word representations in vector space. *CoRR*, abs/1301.3781, 2013.

[15] S. Riezler, A. Vasserman, I. Tsochantaridis, V. Mittal, and Y. Liu. Statistical Machine Translation for Query Expansion in Answer Retrieval. In *Proceedings of ACL'07*, 2007.

[16] M. Surdeanu, M. Ciaramita, and H. Zaragoza. Learning to Rank Answers on Large Online QA Collections. In *Proceedings of ACL'08*, pages 719–727, 2008.

[17] W. tau Yih, M.-W. Chang, C. Meek, and A. Pastusiak. Question answering using enhanced lexical semantic models. In *Proceedings of ACL'13*, 2013.

[18] M. Wang, N. A. Smith, and T. Mitamura. What is the jeopardy model? A quasi-synchronous grammar for QA. In *EMNLP-CoNLL'07*, pages 22–32, 2007.

[19] Q. Wu, C. J. Burges, K. M. Svore, and J. Gao. Adapting Boosting for Information Retrieval Measures. *Inf. Retr.*, 13(3):254–270, June 2010.

[20] X. Xue, J. Jeon, and W. B. Croft. Retrieval Models for Question and Answer Archives. In *Proceedings of SIGIR'08*, pages 475–482, 2008.

[21] X. Yao, B. V. Durme, C. Callison-burch, and P. Clark. Answer extraction as sequence tagging with tree edit distance. In *Proceedings of NAACL'13*, 2013.

[22] L. Yu, K. M. Hermann, P. Blunsom, and S. Pulman. Deep Learning for Answer Sentence Selection. *arXiv:1412.1632 [cs]*, Dec. 2014. arXiv: 1412.1632.