The Use of Metadata, Web-derived Answer Patterns and Passage Context to Improve Reading Comprehension Performance

Yongping Du
Media Computing and Web Intelligence Laboratory
Fudan University
Shanghai, China
ypdu@fudan.edu.cn

Helen Meng
Human-Computer Communication Laboratory
The Chinese University of Hong Kong
HongKong. SAR. China
hmmeng@se.cuhk.edu.hk

Xuanjing Huang
Media Computing and Web Intelligence Laboratory
Fudan University
Shanghai, China
xjhuang@fudan.edu.cn

Lide Wu
Media Computing and Web Intelligence Laboratory
Fudan University
Shanghai, China
ldwu@fudan.edu.cn

Abstract

A reading comprehension (RC) system attempts to understand a document and returns an answer sentence when posed with a question. RC resembles the *ad hoc* question answering (QA) task that aims to extract an answer from a collection of documents when posed with a question. However, since RC focuses only on a single document, the system needs to draw upon external knowledge sources to achieve deep analysis of passage sentences for answer sentence extraction. This paper proposes an approach towards RC that attempts to utilize external knowledge to improve performance beyond the baseline set by the bag-of-words (BOW) approach. Our approach emphasizes matching of metadata (i.e. verbs, named entities and base noun phrases) in passage context utilization and answer sentence extraction. We have also devised an automatic acquisition process for Web-derived answer patterns (AP) which utilizes question-answer pairs from TREC QA, the Google search engine and the Web. This approach gave improved RC performances for both the Remedia and ChungHwa corpora, attaining *HumSent* accuracies of 42% and 69% respectively. In particular, performance analysis based on Remedia shows that relative performances of 20.7% is due to metadata matching and a further 10.9% is due to the application of Web-derived answer patterns.

1. Introduction

A reading comprehension (RC) system attempts to understand a document and returns an answer sentence when posed with a question. The RC task was first proposed by the MITRE Corporation which developed the Deep Read reading comprehension system (Hirschman et al., 1999). Deep Read was evaluated on the Remedia Corpus that contains a set of stories, each with an average of 20 sentences and five questions (of types *who*, *where*, *when*, *what* and *why*). The MITRE group also defined the *HumSent* scoring metric, i.e. the percentage of test questions for which the system has chosen a correct sentence as the answer. *HumSent* answers were compiled by a human annotator, who examined the stories and chose the sentence(s) that best answered the questions. It was judged that for 11% of the Remedia test questions, there is no single sentence in the story that is judged to be an appropriate answer sentence. Hence the upper bound for RC on Remedia should by 89% *HumSent* accuracy. (Hirschman et al. 1999) reported a *HumSent* accuracy of 36.6% on the Remedia test set. Subsequently, (Ng et al., 2000) used a machine learning approach of decision tree and achieved the accuracy of 39.3%. Then (Riloff and Thelen, 2000) and (Charniak et al., 2000) reported improvements to 39.7% and 41%, respectively. They made use of handcrafted heuristics such as the WHEN rule:

if contain(S, TIME), then Score(S)+=4

i.e. WHEN questions reward candidate answer sentences with four extra points if they contain a name entity TIME.

RC resembles the ad hoc question answering (QA) task in TREC.[1] The QA task finds answers to a set of questions from a *collection* of documents, while RC focuses on a single

[1] http://www.nist.gov.

Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing (HLT/EMNLP), pages 604–611, Vancouver, October 2005.

document. (Light et al. 1998) conducted a detailed compared between the two tasks. They found that the answers of most questions in the TREC QA task appear more than once within the document collection. However, over 80% of the questions in the Remedia corpus correspond to answer sentences that have a single occurrence only. Therefore an RC system often has only one shot at finding the answer. The system is in dire need of extensive knowledge sources to help with deep text analysis in order to find the correct answer sentence.

Recently, many QA systems have exploited the Web as a gigantic data repository in order to help question answering (Clarke et al., 2001; Kwok et al., 2001; Radev et al., 2002). Our current work attempts to incorporate a similar idea in exploiting Web-derived knowledge to aid RC. In particular, we have devised an automatic acquisition process for Web-derived answer patterns. Additionally we propose to emphasize the importance of *metadata* matching in our approach to RC. By metadata, we are referring to automatically labeled verbs, named entities as well as base noun phrases in the passage. It is important to achieve a metadata match between the question and a candidate answer sentence before the candidate is selected as the final answer. The candidate answer sentence may be one with a high degree of word overlap with the posed question, or it may come from other sentences in the neighboring context. We apply these different techniques step by step and obtain better results than have ever previously been reported. Especially, we give experiment analysis for understanding the results.

In the rest of this paper, we will first describe three main aspects of our approach towards RC – (i) metadata matching, (ii)automatic acquisition of Web-derived answer patterns and (iii) the use of passage context. This will be followed by a description of our experiments, analysis of results and conclusions.

2. Metadata Matching

A popular approach in reading comprehension is to represent the information content of each question or passage sentence as a bag of words (BOW). This approach incorporates stopword removal and stemming. Thereafter, two words are considered a match if they share the same morphological root. Given a question, the BOW approach selects the passage sentence with the maximum number of matching words as the answer. However, the BOW approach does not capture the fact that the *informativeness* of a word about a passage sentence varies from one word to another. For example, it has been pointed out by (Charniak et al. 2000) that the verb seems to be especially important for recognizing that a passage sentence is related to a specific question. In view of this, we propose a representation for questions and answer sentences that emphasizes three types of metadata:

(i) **Main Verbs (MVerb)**, identified by the link parser (Sleator and Temperley 1993);
(ii) **Named Entities (NE)**, including names of locations (LCN), persons (PRN) and organizations (ORG), identified by a home-grown named entity identification tool; and
(iii) **Base Noun Phrases (BNP)**, identified by a home-grown base noun phrase parser respectively.

We attempt to quantify the relative importance of such metadata through corpus statistics obtained only from the training set of the Remedia corpus, which has 55 stories. The Remedia test set, which contains 60 stories, is set aside for evaluation. On average, each training story has 20 sentences and five questions. There are 274 questions in all in the entire training set. Each question corresponds to a marked answer sentence within the story text. We analyzed all the questions and divided them into three question sets (Q_SETS) based on the occurrences of MVerb, NE and BNP identified with the tools mentioned above. The following are illustrative examples of the Q_SETS as well as their sizes:

Q_SET_{Mverb} (Count:169)	*Who helped the Pilgrims?*
Q_SET_{NE} (Count:62)	*When was the first merry-go-round built in the United States?*
Q_SET_{BNP} (Count:232)	*Where are the northern lights?*

Table 1. Examples and sizes of question sets (Q_SETS) with different metadata – main verb (MVerb), named entity (NE) and base noun phrase (BNP).

It may also occur that a question belongs to multiple Q_SETS. For example:

Q_SET$_{MVerb}$	*When was the first merry-go-round built in the United States?*
Q_SET$_{NE}$	*When was the first merry-go-round built in the United States?*
Q_SET$_{BNP}$	*When was the first merry-go-round built in the United States?*

Table 2. An example sentence that belongs to multiple Q_SETS.

As mentioned earlier, each question corresponds to an answer sentence, which is annotated in the story text by MITRE. Hence we can follow the Q_SETS to divide the answer sentences into three answer sets (A_SETS). Examples of A_SETS that correspond to Table 1 include:

A_SET$_{MVerb}$	*An Indian named Squanto came to help them.*
A_SET$_{NE}$	*The first merry-go-round in the United States was built in 1799.*
A_SET$_{BNP}$	*Then these specks reach the air high above the earth.*

Table 3. Examples of the answer sets (A_SETS) corresponding to the different metadata categories, namely, main verb (MVerb), named entity (NE) and base noun phrase) (BNP).

In order to quantify the relative importance of matching the three kinds of metadata between Q_SET and A_SET for reading comprehension, we compute the following relative weights based on corpus statistics:

$$Weight_{Metadata} = \frac{|S_{Metadata}|}{|A\_SET_{Metadata}|} \quad \text{.....Eqn (1)}$$

where $S_{Metadata}$ is the set of answer sentences in $|A\_SET_{Metadata}|$ that contain the metadata of its corresponding question. For example, referring to Tables 2 and 3, the question in Q_SET$_{NE}$ *"When was the first merry-go-round built in the United Sates?"* contains the named entity (underlined) which is also found in the associated answer sentence from A_SET$_{NE}$, *"The first merry-go-round in the United States was built in 1799." Hence this answer sentence belongs to the set* S$_{NE}$. Contrarily, the question in Q_SET$_{BNP}$ *"Where are the northern lights?"* contains the base noun phrase (underlined) but it is not found in the associated answer sentence from A_SET$_{BNP}$, *"Then these specks reach the air high above the earth."* Hence this answer sentence does not belong to the set S$_{BNP}$. Based on the three sets, we obtain the metadata weights:

$Weight_{MVerb}$=0.64, $Weight_{NE}$=0.38, $Weight_{BNP}$=0.21

To illustrate how these metadata weights are utilized in the RC task, consider again the question, "*Who helped the Pilgrims?*" together with three candidate answers that are "equally good" with a single word match when the BOW approach is applied. We further search for matching metadata among these candidate answers and use the metadata weights for scoring.

Question	*Who helped the Pilgrims?* MVerb identified: "help" BNP identified: "the Pilgrams"
Candidate Sentence 1	*An Indian named Squanto came to help.* Matched MVerb (underlined) Score= $Weight_{MVerb}$=0.64
Candidate Sentence 2	*By fall, the Pilgrims had enough food for the winter.* Matched BNP (underlined) Score= $Weight_{BNP}$=0.21
Candidate Sentence 3	*Then the Pilgrims and the Indians ate and played games.* Matched BNP (underlined) Score= $Weight_{BNP}$=0.21

Table 4. The use of metadata matching to extend the bag-of-words approach in reading comprehension.

3. Web-derived Answer Patterns

In addition to using metadata for RC, the proposed approach also leverages knowledge sources that are external to the core RC resources – primarily the Web and other available corpora. This section describes our approach that attempts to automatically derive answer patterns from the Web as well as score useful answer patterns to aid RC. We utilize the open domain question-answer pairs (2393 in all) from the Question Answering track of TREC (TREC8-TREC12) as a basis for automatic answer pattern acquisition.

3.1 Deriving Question Patterns

We define a set of question tags (Q_TAGS) that extend the metadata above in order to represent *question patterns*. The tags include one for main verbs (Q_MVerb), three for named entities (Q_LCN, Q_PRN and Q_ORG) and one for base noun phrases (Q_BNP). We are also careful to ensure that noun phrases tagged as named entities are not further tagged as base noun phrases.

A question pattern is expressed in terms of Q_TAGS. A question pattern can be used to represent multiple questions in the TREC QA resource. An example is shown in Table 5. Tagging the TREC QA resource provides us with a set of question patterns $\{QP_i\}$ and for each pattern, up to m_i example questions.

Question Pattern (QP_i):
When do Q_PRN Q_MVerb Q_BNP?
Represented questions:
Q_1: *When did Alexander Graham Bell invent the telephone?*
Q_2: *When did Maytag make Magic Chef refrigerators?*
Q_3: *When did Amumdsen reach the South Pole?*
(m_i example questions in all)

Table 5. A question pattern and some example questions that it represents.

3.2 Deriving Answer Patterns

For each question pattern, we aim to derive answer patterns for it automatically from the Web. The set of answer patterns capture possible ways of embedding a *specific* answer in an answer sentence. We will describe the algorithm for deriving answer patterns as following and illustrate with the following question answer pair from TREC QA:

Q: *When did Alexander Graham Bell invent the telephone?*
A: *1876*

1. Formulate the Web Query

The question is tagged and the Web query is formulated as "Q_TAG"+ "ANSWER", i.e.

Question: "*When did Alexander Graham Bell invent the telephone?"*

QP: When do Q_PRN Q_MVerb Q_BNP ?

where Q_PRN= *"Alexander Graham Bell"*, Q_MVerb= "*invent*", and Q_BNP= *"the telephone*"

hence Web query: *"Alexander Graham Bell"+ "invent" + "the telephone" + "1876"*

2. Web Search and Snippet Selection

The Web query is submitted to the search engine Google using the GoogleAPI and the top 100 snippets are downloaded. From each snippet, we select up to ten contiguous words to the left as well as to the right of the "ANSWER" for answer pattern extraction. The selected words must be continuous and do not cross the snippet boundary that Google denotes with '…'.

3. Answer Pattern Selection

We label the terms in each selected snippet with the Q_TAGs from the question as well as the answer tag <A>. The shortest string containing all these tags (underlined below) is extracted as the answer pattern (AP). For example:

Snippet 1: *1876, Alexander Graham Bell invented the telephone in the United States…*
AP 1: <A>, Q_PRN Q_MVerb Q_BNP.
(N.B. The answer tag <A> denotes "*1876*" in this example).

Snippet 2: *…which has been invented by Alexander Graham Bell in 1876…*
AP 2: Q_MVerb by Q_PRN in <A>.

As may be seen in above, the acquisition algorithm for Web-derived answer questions calls for specific answers, such as a factoid in a word or phrase. Hence the question-answer pairs from TREC QA are suitable for use. On the other hand, Remedia is less suitable here because it contains labelled answer *sentences* instead of factoids. Inclusion of whole answer sentences in Web query formulation generally does not return the answer pattern that we seek in this work.

3.3 Scoring the Acquired Answer Patterns

The answer pattern acquisition algorithm returns *multiple* answer patterns for every question-answer pair submitted to the Web. In this subsection we present an algorithm for deriving scores for these answer patterns. The methodology is motivated by the concept of *confidence* level, similar to that used in data mining. The algorithm is as follows:

1. Formulate the Web Query

For each question pattern QP_i (see Table 5) obtained previously, randomly select an example question among the m_i options that belongs to this pattern. The question is tagged and the Web query is formulated in terms of the Q_TAGs only. (Please note that the corresponding *answer is excluded* from Web query formulation here, which differs from the answer pattern acquisition algorithm). E.g.,

Question: "*When did Alexander Graham Bell invent the telephone?*

Q_TAGs: Q_PRN Q_MVerb Q_BNP

Web query: *"Alexander Graham Bell"+ "invent" + "the telephone"*

2. Web Search and Snippet Selection

The Web query is submitted to the search engine

Google and the top 100 snippets are downloaded.

3. Scoring each Answer Pattern AP_{ij} relating to QP_i

Based on the question, its pattern QP_i, the answer and the retrieved snippets, totally the following counts for each answer pattern $\mathbf{AP_{ij}}$ relating to QP_i .

c_{ij} – # snippets matching AP_{ij} and for which the tag <A> matches the correct answer.

n_{ij} – # snippets matching AP_{ij} and for which the tag <A> matches any term

Compute the ratio $r_{ij} = c_{ij} / n_{ij}$.....Eqn(2)

Repeat steps 1-3 above for another example question randomly selected from the pool of m_i example under QP_i. We arbitrarily set the maximum number of iterations to be $k_i = \left\lceil \frac{2}{3} m_i \right\rceil$ in order to achieve decent coverage of the available examples. The confidence for AP_{ij}.is computed as

$$Confidence\ (AP_{ij}) = \frac{\sum_{i=1}^{k} r_{ij}}{k} \ldots\ldots \text{Eqn(3)}$$

Equation (3) tries to assign high confidence values to answer patterns AP_{ij} that choose the correct answers, while other answer patterns are assigned low confidence values. E.g.:

<A>, Q_PRN Q_MVerb Q_BNP (Confidence=0.8)
Q_MVerb by Q_PRN in <A>. (Confidence=0.76)

3.4 Answer Pattern Matching in RC

The Web-derived answer patterns are used in the RC task. Based on the question and its QP, we select the related AP to match among the answer sentence candidates. The candidate that matches the highest-scoring AP will be selected. We find that this technique is very effective for RC as it can discriminate among candidate answer sentences that are rated "equally good" by the BOW or metadata matching approaches, e.g.:

Q: *When is the Chinese New Year?*
QP: When is the Q_BNP?
where Q_BNP=*Chinese New Year*
Related AP: Q_BNP is <A> (Confidence=0.82)
Candidate answer sentences 1: *you must wait a few more weeks for the Chinese New Year.*
Candidate answer sentences 2: *Chinese New Year is most often between January 20 and February 20.*

Both candidate answer sentences have the same number of matching terms – "Chinese", "New" and "Year" and the same metadata, i.e. Q_BNP=*Chinese New Year*. The term "is" is excluded by stopword removal. However the Web-derived answer pattern is able to select the second candidate as the correct answer sentence.

Hence our system gives high priority to the Web-derived AP – if a candidate answer sentence can match an answer pattern with confidence > 0.6, the candidate is taken as the final answer. No further knowledge constraints will be enforced.

4. Context Assistance

During RC, the initial application of the BOW approach focuses the system's attention on a small set of answer sentence candidates. However, it may occur the true answer sentence is not contained in this set. As was observed by (Riloff and Thelen, 2000) and (Charniak et al., 2000), the correct answer sentence often precedes/follows the sentence with the highest number of matching words. Hence both the preceding and following context sentences are searched in their work to find the answer sentence especially for *why* questions.

Our proposed approach references this idea in leveraging contextual knowledge for RC. Incorporation of contextual knowledge is very effective when used in conjunction with named entity (NE) identification. For instance, *who* questions should be answered with words tagged with Q_PRN (for persons). If the candidate sentence with the highest number of matching words does not contain the appropriate NE, it will not be selected as the answer sentence. Instead, our system searches among the *two* preceding and *two* following context sentences for the appropriate NE. Table 6 offers an illustration. Data analysis Remedia training set shows that the context window size selected is appropriate for *when*, *who* and *where* questions.

Football Catches On Fast

(LATROBE, PA., September 4, 1895) - The new game of football is catching on fast, and each month new teams are being formed.

Last night was the first time that a football player was paid. The man's name is John Brallier, and he was paid $10 to take the place of someone who was hurt.…

Question: *Who was the* first *football player to be paid?*
Sentence with maximum # matching words: *Last night was the first time that a football player was paid.*
Correct answer sentence: *The man's name is John Brallier, and he was paid $10 to take the place of someone who was hurt.*

Table 6. An example illustrating the use of contextual knowledge in RC.

As for *why* questions, a candidate answer sentence is selected from the context window if its first word is one of *"this"*, *"that"*, *"these"*, *"those"*, *"so"* or "*because*". We did not utilize contextual constraints for *what* questions.

5. Experiments

RC experiments are run on the Remedia corpus as well as the ChungHwa corpus. The Remedia training set has 55 stories, each with about five questions. The Remedia test set has 60 stories and 5 questions per story. The ChungHwa corpus is derived from the book, *"English Reading Comprehension in 100 days,"* published by Chung Hwa Book Co., (H.K.) Ltd. The ChungHwa training set includes 100 English stories and each has four questions on average. The ChungHwa testing set includes 50 stories and their questions. We use *HumSent* as the prime evaluation metric for reading comprehension.

The three kinds of knowledge sources are used incrementally in our experimental setup and results are labeled as follows:

Result	Technique
Result_1	BOW
Result_2	BOW+MD
Result_3	BOW+MD+AP
Result_4	BOW+MD+AP+Context

Table 7. Experimental setup in RC evaluations. Abbrievations are: bag-of-words (BOW), metadata (MD), Web-derived answer patterns (AP), contextual knowledge (Context).

5.1 Results on Remedia

Table 8 shows the RC results for various question types in the Remedia test set.

	When	Who	What	Where	Why
Result_1	32.0%	30.0%	31.8%	29.6%	18.6%
Result_2	40.0%	28.0%	39.0%	38.0%	20.0%
Result_3	52.6%	42.8%	40.6%	38.4%	21.0%
Result_4	55.0%	48.0%	40.6%	36.4%	27.6%

Table 8. *HumSent* accuracies for the Remedia test set.

We observe that the *HumSent* accuracies vary substantially across different interrogatives. The system performs best for *when* questions and worst for *why* questions. The use of Web-derived answer patterns brought improvements to all the different interrogatives. The other knowledge sources, namely, meta data and context, bring improvements for some question types but degraded others.

Figure 1 shows the overall RC results of our system. The relative incremental gains due to the use of metadata, Web-derived answer patterns and context are 20.7%, 10.9% and 8.2% respectively. We also ran pairwise *t*-tests to test the statistical significance of these improvements and results are shown in Table 9. The improvements due to metadata matching and Web-derived answer patterns are statistically significant ($p<0.05$) but the improvement due to context is not.

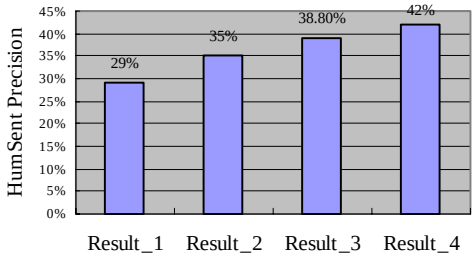

Figure 1. *HumSent* accuracies for Remedia.

Pairwise Comparison	Result_1 & Result_2	Result_2 & Result_3	Result_3 & Result_4
t-test Results	t(4)=2.207, p=0.046	t(4)=2.168, p=0.048	t(4)=1.5, p=0.104

Table 9. Tests of statistical significance in the incremental improvements over BOW among the use of metadata, Web-derived answer patterns and context.

We also compared our results across various interrogatives with those previously reported in (Riloff and Thelen, 2000). Their system is based on handcrafted rules with deterministic algorithms. The comparison (see Table 10) shows that our approach which is based on data-driven patterns and statistics can achieve comparable performance.

Question Type	Riloff &Thelen 2000	Result_4
When	55%	55.0%
Who	41%	48.0%
What	28%	40.6%
Where	47%	36.4%
Why	28%	27.6%
Overall	**40%**	**42.0%**

Table 10. Comparison of *HumSent* results with a heuristic based RC system (Riloff & Thelen 00).

5.2 Results on ChungHwa

Experimental results for the ChungHwa corpus are presented in Figure 2. The *HumSent* accuracies obtained are generally higher than those with

Remedia. We observe similar trends as before, i.e. our approach in the use of metadata, Web-derived answer patterns and context bring incremental gains to RC performance. However, the actual gain levels are much reduced.

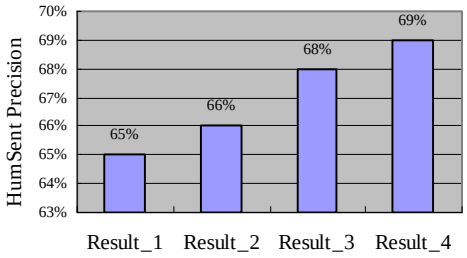

Figure 2. *HumSent* accuracies for ChungHwa.

5.3. Analyses of Results

In order to understand the underlying reason for reduced performance gains as we migrated from Remedia to Chunghwa, we analyzed the question lengths as well as the degree of word match between questions and answers among the two corpora. Figure 3 shows that the average length of questions in Chunghwa are longer than Remedia. Longer questions contain more information which is beneficial to the BOW approach in finding the correct answer.

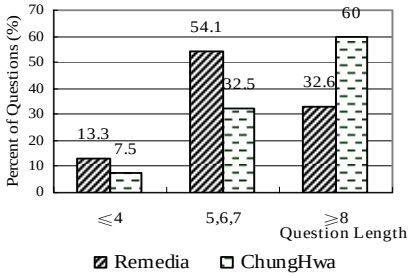

Figure 3. Distribution of question lengths among the Remedia and ChungHwa corpora.

The degree of word match between questions and answers among the two corpora is depicted in Figure 4. We observe that ChungHwa has a larger proportion of questions that have a *match- size* (i.e. number of matching words between a question and its answer) larger than 2. This presents an advantage for the BOW approach in RC. It is also observed that approximately 10% of the Remedia questions have no correct answers (i.e. match-size=-1) and about 25% have no matching words with the correct answer sentence. This explains the overall discrepancies in *HumSent* accuracies between Remedia and ChungHwa.

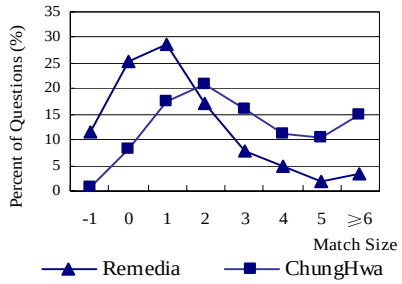

Figure 4. Distribution of match-sizes (i.e. the number of matching words between questions and their answers) in the two corpora.

While our approach has leveraged a variety of knowledge sources in RC, we still observe that our system is unable to correctly answer 58% of the questions in Remedia. An example of such elusive questions is:

Question: *When do the French celebrate their freedom?*
Answer Sentence: *To the French, July 14 has the same meaning as July 4th does to the United States.*

6. Conclusions

A reading comprehension (RC) system aims to understand a single document (i.e. story or passage) in order to be able to automatically answer questions about it. The task presents an information retrieval paradigm that differs significantly from that found in Web search engines. RC resembles the question answering (QA) task in TREC which returns an answer for a given question from a collection of documents. However, while a QA system can utilize the knowledge and information in a *collection* of documents, RC systems focuses only on a *single* document only. Consequently there is a dire need to draw upon a variety of knowledge sources to aid deep analysis of the document for answer generation. This paper presents our initial effort in designing an approach for RC that leverages a variety of knowledge sources beyond the context of the passage, in an attempt to improve RC performance beyond the baseline set by the bag-of-words (BOW) approach. The knowledge sources include the use of metadata (i.e. verbs, named entities and base noun phrases). Metadata matching is applied in our approach in answer sentence extraction as well as use of contextual sentences. We also devised an

automatic acquisition algorithm for Web-derived answer patterns. The acquisition process utilizes question-answer pairs from TREC QA, the Google search engine and the Web. These answer patterns capture important structures for answer sentence extraction in RC. The use of metadata matching and Web-derived answer patterns improved reading comprehension performances for the both Remedia and ChungHwa corpora. We obtain improvements over previously reported results for Remedia, with an overall HumSet accuracy of 42%. In particular, a relative gain of 20.7% is due to metadata matching and a further 10.9% is due to application of Web-derived answer patterns.

Acknowledgement

This work is partially supported by the Direct Grant from The Chinese University of Hong Kong (CUHK) and conducted while the first author was visiting CUHK. This work is supported by Natural Science Foundation of China under Grant No.60435020.

References

Charles L.A. Clarke, Gordon V. Cormack, Thomas R. Lynam. 2001. Exploiting Redundancy in Question Answering. In Proceedings of the 24th ACM Conference on Research and Development in Information Retrieval (SIGIR-2001, New Orleans, LA). ACM Press. New York, 358–365.

Cody C. T. Kwok, Oren Etzioni, Daniel S. Weld. 2001. Scaling Question Answering to the Web. In Proceedings of the 10th World Wide Web Conference (WWW'2001). 150-161.

Daniel Sleator and Davy Temperley. 1993. Parsing English with a Link Grammar. Third International Workshop on Parsing Technologies.

Deepak Ravichandran and Eduard Hovy. 2002. Learning Surface Text Patterns for a Question Answering System. In Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL-2002). 41-47.

Dell Zhang, Wee Sun Lee. 2002. Web Based Pattern Mining and Matching Approach to Question Answering. In Proceedings of the TREC-11 Conference. 2002. NIST, Gaithersburg, MD, 505-512.

Dragomir Radev, Weiguo Fan, Hong Qi, Harris Wu, Amardeep Grewal. 2002. Probabilistic Question Answering on the Web. In Proceedings of the 11th World Wide Web Conference (WWW'2002).

Ellen Riloff and Michael Thelen. 2000. A Rule-based Question Answering System for Reading Comprehension Test. ANLP/NAACL-2000 Workshop on Reading Comprehension Tests as Evaluation for Computer-Based Language Understanding Systems.

Eugene Charniak, Yasemin Altun, Rodrigo de Salvo Braz, Benjamin Garrett, Margaret Kosmala, Tomer Moscovich, Lixin Pang, Changhee Pyo, Ye Sun, Wei Wy, Zhongfa Yang, Shawn Zeller, and Lisa Zorn. 2000. Reading Comprehension Programs in a Statistical-Language-Processing Class. ANLP-NAACL 2000 Workshop: Reading Comprehension Tests as Evaluation for Computer-Based Language Understanding Systems.

Hwee Tou Ng, Leong Hwee Teo, Jennifer Lai Pheng Kwan. 2000. A Machine Learning Approach to Answering Questions for Reading Comprehension Tests. Proceedings of the 2000 Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora 2000.

Lynette Hirschman, Marc Light, Eric Breck, and John Burger. 1999. Deep Read: A Reading Comprehension System. Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics.

Marc Light, Gideon S. Mann, Ellen Riloff and Eric Break. 1998. Analyses for Elucidating Current Question Answering Technology. Natural Language Engineering. Vol. 7, No. 4.

Martin M. Soubbotin, Sergei M. Soubbotin. 2002. Use of Patterns for Detection of Likely Answer Strings: A Systematic Approach. In Proceedings of the TREC-11 Conference. 2002. NIST, Gaithersburg, MD, 134-143.