Computational Linguistics and Chinese Language Processing
Vol. 13, No. 1, March 2008, pp. 1-26

Exploring Shallow Answer Ranking Features in Cross-Lingual and Monolingual Factoid Question Answering

Cheng-Wei Lee[*+], Yi-Hsun Lee[+], and Wen-Lian Hsu[+]

Abstract

Answer ranking is critical to a QA (Question Answering) system because it determines the final system performance. In this paper, we explore the behavior of shallow ranking features under different conditions. The features are easy to implement and are also suitable when complex NLP techniques or resources are not available for monolingual or cross-lingual tasks. We analyze six shallow ranking features, namely, *SCO-QAT*, *keyword overlap*, *density*, *IR score*, *mutual information score*, and *answer frequency*. SCO-QAT (Sum of Co-occurrence of Question and Answer Terms) is a new feature proposed by us that performed well in NTCIR CLQA. It is a co-occurrence based feature that does not need extra knowledge, word-ignoring heuristic rules, or special tools. Instead, for the whole corpus, SCO-QAT calculates co-occurrence scores based solely on the passage retrieval results. Our experiments show that there is no perfect shallow ranking feature for every condition. SCO-QAT performs the best in C-C (Chinese-Chinese) QA, but it is not a good choice in E-C (English-Chinese) QA. Overall, Frequency is the best choice for E-C QA, but its performance is impaired when translation noise is present. We also found that passage depth has little impact on shallow ranking features, and that a proper answer filter with fined-grained answer types is important for E-C QA. We measured the performance of answer ranking in terms of a newly proposed metric EAA (Expected Answer Accuracy) to cope with cases of answers that have the same score after ranking.

[*] Department of Computer Science, National Tsing-Hua University, Taiwan, R.O.C, 101, Section 2, Kuang-Fu Road, Hsinchu, Taiwan, R.O.C.

[+] Institute of Information Science, Academia Sinica, Taiwan, R.O.C, 128 Academia Road, Section 2, Nankang, Taipei 115, Taiwan, R.O.C.
The author for correspondence is Wen-Lian Hsu.
E-mail: {aska, rog, hsu}@iis.sinica.edu.tw

[Received June 30, 2007; Revised October 8, 2007; Accepted January 12, 2008]

Keywords: Answer Ranking, Co-occurrence, CLQA, Question Answering, Shallow Method, SCO-QAT

1. Introduction

In recent years, question answering (QA) has become a key research area in several of the world's major languages, possibly because of the urgent need to deal with the information overload caused by the rapid growth of the Internet. Since 1999, many international question answering contests have been held at conferences and workshops, such as TREC[1], CLEF[2], and NTCIR[3]. Thus far, several languages – such as Bulgarian, Dutch, English, Finnish, French, German, Indonesian, Italian, Japanese, Portuguese, and Spanish – have been tested in monolingual or cross-lingual question answering tasks. In QA research, questions are usually classified into several categories, such as factoid questions, list questions, and definition questions, then dealt with by different techniques. Among these categories, factoid questions have been studied the most widely, and they are the focus of this paper.

There is usually exactly one answer, which is a noun or short phrase, for a factoid question. For example, "Who is the president of the United States?" is a factoid question because the name of the president is a noun, and there is only one current U.S. President. Factoid questions are usually classified into questions types, such as Q_PERSON, Q_LOCATION, Q_ORGANIZATION, Q_ARTIFACT, Q_TIME, and Q_NUMBER [Lee *et al*. 2007; Lee *et al.* 2005]. Although question types vary in different contests and different systems, the corresponding answer types can usually be recognized by named entity recognition (NER) techniques or simple rules.

A QA system is normally comprised of several modules. The answer ranking module implements the last step in answering a factoid question and determines the final performance. After candidate answers have been extracted from retrieved passages, the answer ranking module takes the question, the passages (or documents), and the candidate answers as input, ranks the candidate answers, and then outputs a ranked list of candidate answers. Although several answer ranking methods have been proposed, they can be generally categorized as either deep or shallow methods. A deep method uses complex NLP techniques and may require extensive rules, ontologies, or human effort, while a shallow method does not require much of these resources and is therefore cheaper to implement.

Although deep answer ranking methods have proven useful for English QA, as reported in [Cui *et al*. 2005; Harabagiu *et al*. 2005], the resources needed for such methods are usually

[1] Text REtrieval Conference (TREC). http://trec.nist.gov/

[2] Cross-Language Evaluation Forum (CLEF). http://www.clef-campaign.org/

[3] NTCIR (NII Test Collection for IR Systems) Project. http://research.nii.ac.jp/ntcir/

not available for some languages in monolingual or cross-lingual QA. In those cases, shallow ranking methods have to be used; however, to the best of our knowledge, very little research has been done on such methods. The situation is worse for cross-lingual tasks because most cross-lingual QA research has focused on the front-end modules, *i.e.*, question processing and passage retrieval. Research on back-end modules, such as answer ranking, has received little attention in the cross-lingual QA domain.

In this paper, we attempt to fill this research gap by exploring the behavior of shallow ranking features under noise produced by other QA modules in both monolingual and cross-lingual situations. Herein, noise is defined in terms of the performance decrement of a QA module. For example, in the case of translation quality decrement, we say that we encounter translation noise and expect that the noise may impact the performance of some shallow ranking features. In addition to translation noise, we also consider passage retrieval noise and answer filter noise. We measure the influence of these types of noise by three performance metrics to determine which ranking feature is the most effective in dealing with each kind of noise.

Apart from considering widely used shallow ranking features, we propose a new ranking feature called SCO-QAT, which has been successfully applied to the ASQA2 system [Lee *et al*. 2007], and also achieved the best performance on the C-C and E-C subtasks in NTCIR-6 CLQA [Sasaki *et al*. 2007]. SCO-QAT is a co-occurrence based feature; however, unlike some co-occurrence features [Magnini *et al*. 2001], it does not need extra knowledge, word-ignoring heuristic rules, or special tools.

The remainder of this paper is organized as follows. Related works are discussed in Section 2. We introduce the SCO-QAT feature in Section 3. The evaluation metrics used are introduced in Section 4. The ASQA2 system used in our experiments is described in Section 5. We detail our experiment results and compare SCO-QAT with other shallow features in Section 6. Then, we present our conclusions in Section 7.

2. Related Work

Answer Ranking approaches can be divided in to deep and shallow methods. Deep approaches involve sophisticated tools or knowledge. The most advanced deep methods are logic-based and dependency-parser-based. The LCC team [Harabagiu *et al*. 2005] used an abductive inference method to evaluate the correctness of an answer according to the logic form of the question, the logic form of the sentence that supports the answer, and background knowledge from WordNet. The logic-based approach has achieved the best QA performance in TREC for several years.

Dependency-parser-based methods have also performed quite well on TREC tasks. The National University of Singapore team [Cui *et al.* 2005] used dependency relations identified by a dependency parser to select answer nuggets for factoid and list questions. The similarity between the question and the supporting passage is calculated by machine translation models. Shen [Shen *et al.* 2006] also used dependency relations, but incorporated them into a Maximum Entropy-based ranking model.

Although these deep approaches perform well on monolingual QA (about 0.7 accuracy), they are quite demanding in terms of linguistic resources and computational complexity. In cross-lingual or multilingual QA, it is usually impossible to employ deep approaches for some languages due to the lack of knowledge resources or tools. In contrast, approaches with shallow features are much more flexible when QA languages are changed. The following are some commonly used shallow approaches.

Surface patterns [Soubbotin and Soubbotin 2001] have been successful in the TREC QA Track, which uses string patterns to match questions with correct answers. However, from our perspective, if surface patterns are manually created, the method can not be regarded as "shallow", because it is likely labor intensive. Although there are some "shallow" variations [Geleijnse and Korst 2006; Ravichandran and Hovy 2002] that attempt to create surface patterns automatically/semi-automatically, they usually suffer from the low coverage problem, which means they can only be applied to a few questions.

Some approaches focus on local information, thus only take the *similarity* between a passage and the question into account when finding relevant answers. The simplest way to measure the similarity is by counting the ratio of question terms occurring in the answer passage, as has been reported [Cooper and Ruger 2000; Molla and Gardiner 2005; Zhao *et al.* 2005]. Kwok [Kwok and Deng 2006] and AnswerBus [Zheng 2002] adopt the IR score of the answer passage directly as a measure of similarity. Intuitively, the closeness of two terms may indicate a relation; therefore, some systems [Gillard *et al.* 2006; Lin *et al.* 2005; Lin *et al.* 2005; Sacaleanu and Neumann 2006; Tom´as *et al.* 2005] use features based on the distance between the answer and the question terms to obtain a better similarity measurement. Among these approaches, those of Lin *et al.* [Lin *et al.* 2005] and Roussinov *et al.* [Roussinov *et al.* 2004] incorporate the IDF value with term distances. The assumption is that, if the candidate answer is close to several keywords or question terms, it is more likely to be relevant.

Instead of utilizing local information, which only considers the question and a passage, *redundancy-based* features consider all the returned passages or the entire corpus. Clarke [Clarke *et al.* 2001] suggested that redundancy could be used as a substitute for deep analysis because correct answers may appear many times in high-ranking passages. Features using frequency or co-occurrence information are all regarded as redundancy-based. Several systems [Clarke *et al.* 2002; Cooper and Ruger 2000; Kwok and Deng 2006; Lin *et al.* 2005; Zhao *et al.*

2005; Zheng 2002] include answer frequency in their Answer Ranking components. A web-based co-occurrence shallow feature developed by Magnini *et al*. [Magnini *et al*. 2001] has been successfully applied on the TREC dataset. Magnini used three methods, *Pointwise Mutual Information*, *Maximal Likelihood Ratio*, and *Corrected Conditional Probability*, to measure the co-occurrence of each answer and the given question based on Web search results. However, to use Magnini's method, we also need some word-ignoring heuristic rules to remove search keywords when the number of returned web pages is insufficient.

3. The SCO-QAT Ranking Feature

Before comparing shallow ranking features, we define the SCO-QAT ranking feature that was applied successfully in the ASQA2 system at NTCIR-6. SCO-QAT relies on co-occurrence information about question terms and answer terms, and is therefore similar to Magnini's approach [Magnini *et al*. 2001]. However, unlike Magnini's approach, which utilizes the Web as a corpus to help answer questions posed on a local corpus, SCO-QAT uses passages retrieved by the passage retrieval module from the local corpus directly and does not use any word-ignoring rules.

The basic assumption of SCO-QAT is that, with good quality passages, the more often an answer co-occurs with question terms, the higher the probability that it is correct. Next, we describe the SCO-QAT function. Let the given answer be A and the given question be Q, where Q consists of a set, QT, of question terms {qt_1, qt_2, qt_3, ……, qt_n}. Based on QT, we define QC as a set of question term combinations, or more precisely {qc_i | qc_i is a subset of QT and qc_i is not empty}. We also define a *freq(X)* function of a set X to indicate the number of retrieved passages in which all elements of X co-occur. The relation confidence is calculated as:

$$Conf(qc_i, A) = \begin{cases} \dfrac{freq(qc_i, A)}{freq(qc_i)}, & \text{if } freq(qc_i) \neq 0 \\ 0, & \text{if } freq(qc_i) = 0 \end{cases} . \quad (1)$$

Then, the SCO-QAT formula is defined as:

$$SCO\text{-}QAT(A) = \sum_{i=1}^{|QC|} Conf(qc_i, A) . \quad (2)$$

For example, given a question Q consisting of three question terms {qt1, qt2, qt3} and a corresponding answer set {c1, c2}, the retrieved passages are presented as follows:

P1: qt1 qt2 c2

P2: qt1 qt2 qt3 c1

P3: qt1 qt2 c1

P4: qt1 c2

P5: qt2 c2

P6: qt1 qt3 c1 .

We use Equation (2) to calculate the candidate answer's SCO-QAT score as follows:

$$SCO\text{-}QAT(c1) = \frac{freq(qt1,c1)}{freq(qt1)} + \frac{freq(qt2,c1)}{freq(qt2)} + \frac{freq(qt3,c1)}{freq(qt3)} + \frac{freq(qt1,qt2,c1)}{freq(qt1,qt2)}$$
$$+ \frac{freq(qt1,qt3,c1)}{freq(qt1,qt3)} + \frac{freq(qt2,qt3,c1)}{freq(qt2,qt3)} + \frac{freq(qt1,qt2,qt3,c1)}{freq(qt1,qt2,qt3)}$$
$$= \frac{3}{5} + \frac{2}{4} + \frac{2}{2} + \frac{2}{3} + \frac{2}{2} + \frac{1}{1} + \frac{1}{1} \quad = \quad 5.77$$
$$SCO\text{-}QAT(c2) = \frac{2}{5} + \frac{2}{4} + \frac{0}{2} + \frac{1}{3} + \frac{0}{2} + \frac{0}{1} + \frac{0}{1} \;=\; 1.23 \quad .$$

Since the SCO-QAT score of c1 is higher than that of c2, c1 is considered a better answer candidate than c2.

The rationale behind SCO-QAT is that we try to use retrieved passages as a resource to look up question terms and locate the correct answer. When a set of question terms QT co-occurs with an answer A, we can infer that some kind of relation exists between the QT set and the answer A, which could be helpful for identifying correct answers. However, as this kind of relation is not always correct, we have to find a way to deal with noisy relations. To this end, we use the confidence score shown in Equation (1) to measure the goodness of a rule, which is similar to the method used for finding association rules. Then, we take the sum of the confidence scores of all the co-occurrences of all question term combinations to resolve the noisy rule problem. This technique is useful if the returned passages contain a lot of redundant information about the given question and the answer.

4. Evaluation Metrics

In this section, we describe the evaluation metrics used in this paper.

R-Accuracy and RU-Accuracy

Two metrics, R-Accuracy and RU-Accuracy, are used to measure QA performance in NTCIR CLQA. A QA system returns a list of ranked answer responses for each question, but R-accuracy and RU-accuracy only consider the correctness of the top-1 ranked answer response on the list. An answer response is a pair comprised of an answer and its source document. Each answer response is judged as Right, Unsupported, or Wrong, as defined in the

NTCIR-6 CLQA overview [Lee *et al*. 2007]:

"Right (R): the answer is correct and the source document supports it.

Unsupported (U): the answer is correct, but the source document cannot support it as a correct answer. That is, there is insufficient information in the document for users to confirm by themselves that the answer is the correct one.

Wrong (W): the answer "is incorrect."

Based on these criteria, the accuracy is calculated as the number of correctly answered questions divided by the total number of questions. R-accuracy means that only "Right" judgments are regarded as correct, while RU-accurakcy means that both "Right" and "Unsupported" judgments are counted. As R-accuracy only occurs a few times in this paper, we use "accuracy" to refer to RU-accuracy when the context is not ambiguous.

$$R-Accuracy=\frac{\text{the number of questions for which the top1 rank answer is Right}}{\text{number of questions}}$$

$$RU-Accuracy=\frac{\text{the number of questions for which the top1 rank answer is Right or Unsupported}}{\text{number of questions}}$$

Mean Reciprocal Rank (MRR)

We use MRR when we want to measure QA performance based on all the highest ranked correct answers, not only the top1 answer. MRR is calculated as follows:

$$MRR=\frac{1}{\text{number of questions}}\sum_{question_i}\begin{cases}\dfrac{1}{\text{the highest rank of correct answers}}, & \text{if a correct answer exists}\\ 0, \text{if no correct answer}\end{cases}$$

Expected Answer Accuracy (EAA)

In addition to using the normal answer accuracy metrics, we propose a new metric called the Expected Answer Accuracy (EAA). We use EAA for cases where there are several top answers with the same ranking score.

The EAA score of a ranking method is defined as follows:

$$EAA=\frac{1}{\text{number of questions}}\sum_{question_i}\frac{\text{number of correct answers with top1 rank score}}{\text{number of answers with top1 rank score}}$$

Translation Cost

We use the "translation cost" metric to measure the cost of introducing the cross-lingual function to a QA system. It is calculated as follows:

$$TranslationCost=\frac{\text{accuracy of crosslingual QA - accuracy of monolingual QA}}{\text{accuracy of monolingual QA}}$$

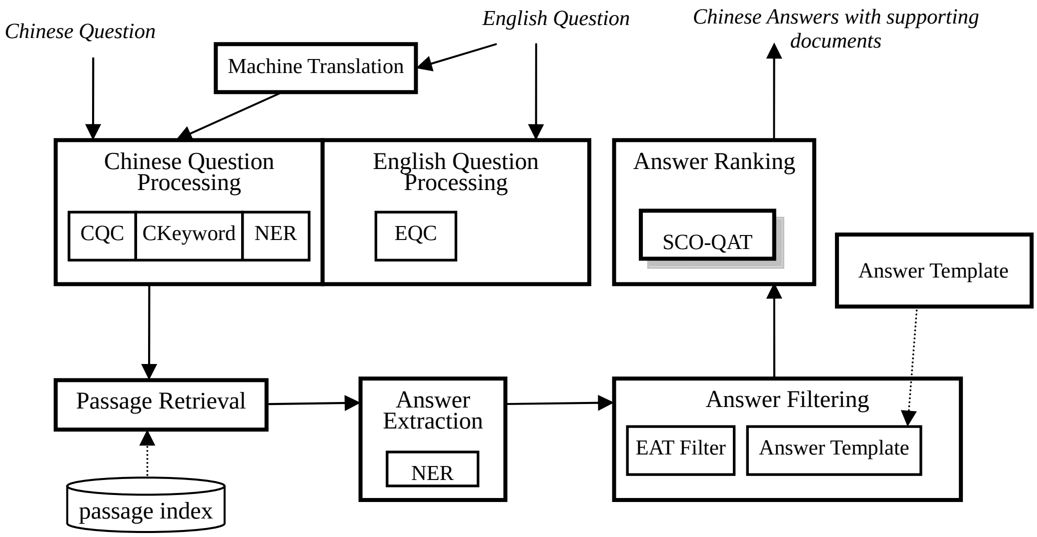

Figure 1. System architecture of ASQA2 for Chinese-Chinese and English-Chinese Factoid QA

5. The Testbed System: the ASQA2 Question Answering System

To evaluate answer ranking features, we chose the Academia Sinica Question Answering (ASQA) system as the testbed system for our experiment because it is modular and it performs well. Moreover, we can easily input different types of noise by adjusting the QA modules in ASQA. The system was developed by Academia Sinica[4] to deal with Chinese related QA tasks. The first version, ASQA1, can only deal with C-C QA, though. ASQA2, which is an extension of ASQA1, can deal with both C-C and E-C QA. We used ASQA1 in NTCIR-5 CLQA and ASQA2 in NTCIR-6 CLQA. NTCIR CLQA is the only QA contest in the world that focuses on Asian languages.

On the C-C and E-C subtasks in NTCIR-6 CLQA, ASQA2 achieved the best performance with 0.553 and 0.34 RU-Accuracy, respectively. The system consists of several modules, as shown in Figure 1. In Question Processing, ASQA2 uses SVMs (Support Vector Machines) and syntax rules to identify the input question type and infer the expected answer types. The type taxonomy has 6 coarse-grained and 62 fined-grained answer types. For passage retrieval, we use Lucene[5], an open source IR engine. The passage depth (the largest number of passages returned by the Passage Retrieval module) for each question is 100. Answers are then extracted from the returned passages by a fined-grained NER engine, and

[4] Academia Sinica, http://www.sinica.edu.tw

[5] Lucene, http://lucene.apache.org/

filtered by the Answer Filtering module according to the question type, answer type, and a mapping table that defines the types' compatibility. The final input for Answer Ranking is comprised of the question, the retrieved passages, and a set of filtered answers. Several answer ranking features are combined as a weighted sum. To deal with cross-lingual QA, ASQA2 adopts the *question translation* approach. Questions are translated with off-the-shelf machine translation engines.

Normally, a cross-lingual QA system is constructed by modifying some components of a monolingual system; however, since translation is involved, the approach often results in performance deterioration. The degree of performance deterioration is usually used with the accuracy metric to evaluate the effectiveness of a cross-lingual system. We define the performance deterioration in terms of the translation cost, which is defined in Section 4. Figure 2 shows the *translation cost* of systems in NTCIR-6 CLQA. When measuring the RU-Accuracy, the translation cost of ASQA2 ranks third, only slightly lower than the system in second place. Therefore, we consider that ASQA2 is an acceptable platform for our mono-lingual and cross-lingual experiments.

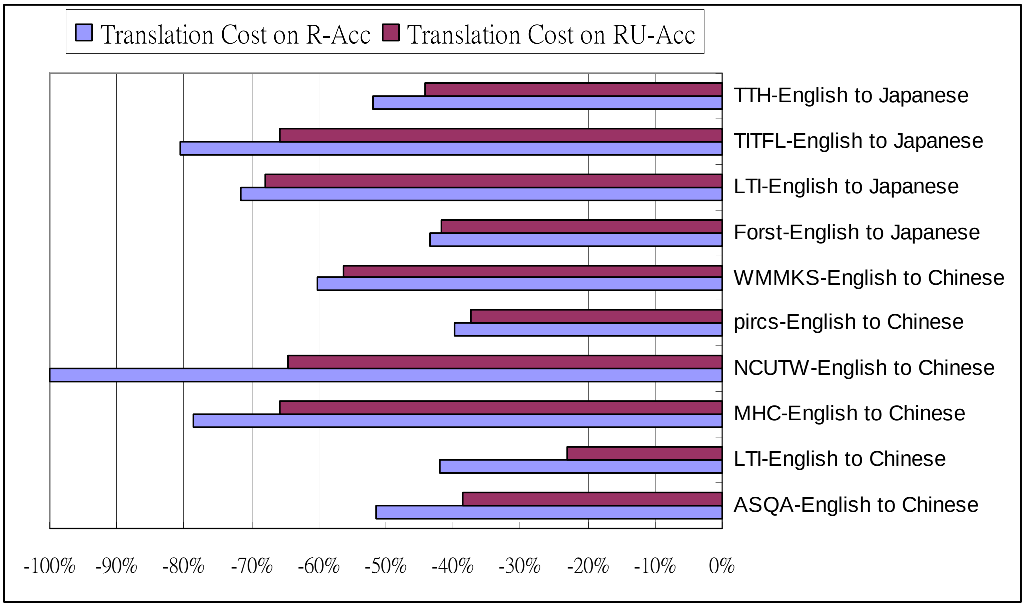

Figure 2. Translation costs of NTCIR-6 CLQA systems for factoid questions. The translation cost is calculated as the performance difference between cross-lingual and mono-lingual systems, divided by the mono-lingual performance.

According to the ASQA2 working notes [Lee *et al*. 2007], the system's success is attributable to three techniques: English question classification, answer template-based answer filtering, and answer ranking with the SCO-QAT feature. When the answer template-based answer filter is applied, it removes all the candidates except the one it deems correct. As it is impossible to compare ranking methods when there is only one answer, we removed the answer template-based filter so that it would not influence our analysis of the answer ranking features.

6. Experiments

We conducted four experiments to explore the behavior of SCO-QAT and other shallow ranking features. In Experiment 1, we observed how shallow ranking features perform when a monolingual QA system is extended to a cross-lingual system. In Experiments 2, 3, and 4, we simulated situations where noise is introduced from the front-end modules and tried to determine which ranking feature is the most suitable under each kind of noise.

6.1 Variable Dependencies

Our testbed system is composed of several modules. Having described the system architecture in Section 5, we now elaborate on the dependencies between the experimental variables. First, we analyze the testbed system to identify several experimental variables and determine their interdependency, as shown in Figure 3. We are interested in the variables in bold font, as they will be used as independent or dependent variables in our experiments. The variables in gray font are not of interest because they are always controlled in the experiments. We provide details of the interdependency of the variables next.

In this study, we focus on the *Accuracy* and other QA performance metrics; therefore, they are always dependent variables. These performance metrics are directly influenced by three variables: the *ranking feature*, *passage quality*, and *answer quality*, since ranking features can use passages and answers. Furthermore, *passage quality* depends on the information retrieval model (*IR model*) used and the *passage depth* (the number of passages used for answer extraction). The greater the *passage depth*, the worse the passage quality is likely to be, which could result in more answers of progressively lower quality.

When ASQA switches from a monolingual to cross-lingual task, two variables are triggered: *translation* and *English question classification*. When translation is active, a translation engine has to be chosen to translate the question. Bad translation quality has a chain reaction effect because it leads to bad query quality, which leads to bad *passage quality* and bad *answer quality*. In ASQA, answer extraction is based on named entity recognition (NER) and answer filtering is based on the compatibility of the question type and the answer type. Therefore, NER and *question classification* are two more variables that could influence *answer quality*.

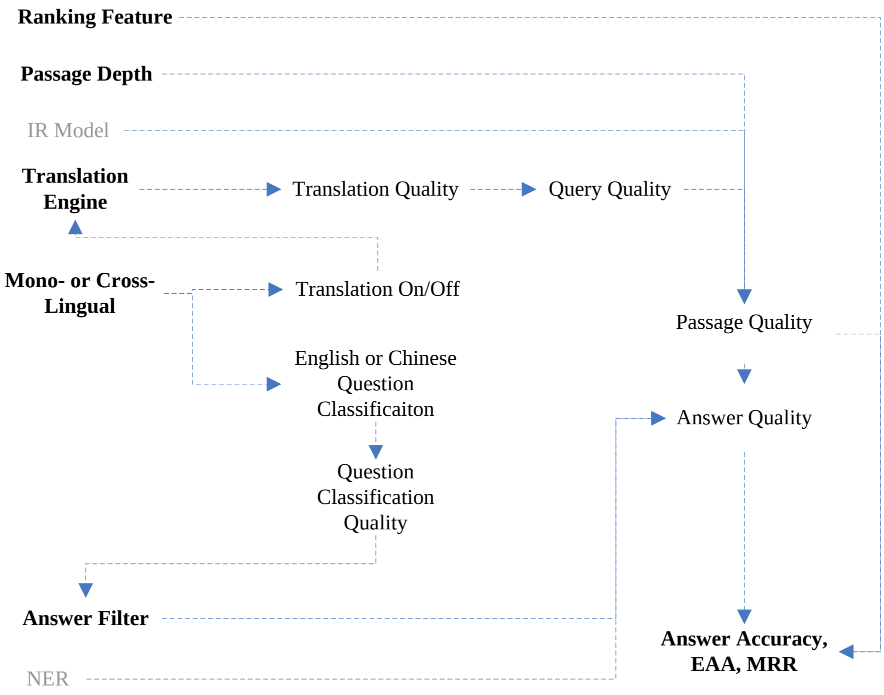

Figure 3. Dependencies of experimental variables based on the architecture of ASQA 2. When a variable at the tail of an arrow changes, it would have influence on the variable at the arrow head.

6.2 QA Datasets

We experimented on several QA datasets. A QA dataset is comprised of a set of questions, their answers, and the document IDs of supporting documents. The answers and supporting documents are regarded as the gold standard. We used the following six datasets from NTCIR5 and NTCIR6 for the CLQA Chinese-Chinese (CC) and English-Chinese (EC) subtasks: NTCIR5-CC-D200, NTCIR5-CC-T200, NTCIR5-EC-D200, NTCIR5-EC-T200, NTCIR6-CC-T150, and NTCIR6-EC-T150. The last item of a dataset name indicates the number of questions and the dataset's purpose, where T stands for "test" and D stands for "development". The CIRB40 corpus was used to compile the NTCIR5 CLQA datasets. It contains 901,446 Chinese newspaper news items published in 2000 and 2001. The corpus used for NTCIR6 CLQA was CIRB20, and it contains 249,508 Chinese newspaper news items published in 1998 and 1999.

Table 1. Datasets for experiments in this paper. Datasets created by NTCIR also has corresponding expanded datasets which consist of extra answer for post-hoc experiment. We postfix a "e" letter to the original name as the name of the expanded dataset name.

	corpus	question amount	creator	languages
NTCIR5-CC-D200	CIRB40	200	NTCIR	C-C
NTCIR5-CC-T200	CIRB40	200	NTCIR	C-C
NTCIR6-CC-T150	CIRB20	150	NTCIR	C-C
IASL-CC-Q465	CIRB40	465	Academia Sinica	C-C
		1015		
NTCIR5-EC-D200	CIRB40	200	NTCIR	E-C
NTCIR5-EC-T200	CIRB40	200	NTCIR	E-C
NTCIR6-EC-T150	CIRB20	150	NTCIR	E-C
		550		

According to Lin [Lin 2005], datasets created by QA evaluation forums are not suitable for post-hoc evaluation because the gold standard is not sufficiently comprehensive. This means we have to manually check all the extra answers not covered by the gold standard in order to derive more reliable experiment results. Since the number of questions in our experiments is quite large, it is not feasible for us to examine all the extra answers and their supporting documents. Therefore, we only use RU-accuracy to compare performances so that we do not have to check all the returned documents; only the answers are checked. These manually examined answers are then fed back to the datasets to form six expanded datasets: NTCIR5-CC-D200e, NTCIR5-CC-T200e, NTCIR5-EC-D200e, NTCIR5-EC-T200e, NTCIR6-CC-T150e, and NTCIR6-EC-T150e. In addition, we created the IASL-CC-Q465 dataset to increase the degree of confidence in our experiments. It was developed by three people using a program that randomly selected passages from the CIRB40 corpus, searched for relevant documents, and created questions from the collected documents. Finally, we had 1015 questions for the C-C task and 550 questions for the E-C task.

6.3 Experiment 1 – Single Shallow Features

Answer correctness features are usually combined in order to achieve the best performance. However, combining features in QA relies mostly on heuristic methods. Although some systems use machine learning approaches successfully for QA ranking, it is rare to see the same approach being applied to other QA work. This may be because QA feature combination methods are not mature enough to deal with the variability of QA systems, and the amount of training data is not sufficient to train good models. Therefore, instead of combined features, we only studied the effect of single ranking features because we assume they are more reliable and can be easily applied to other systems or languages. Table 2 shows the experimental set-up.

Table 2. Experimental Set-up for Experiment 1 – Single Shallow Features

Independent Variables	Ranking Feature, Mono- or Cross-lingual
Dependent Variables	Accuracy, MRR, EAA
Controlled Variables	Passage Depth, Translation Engine, Answer Filter

Along with SCO-QAT, we tested the following widely used shallow features: *keyword overlap* (KO), *density*, *IR score* (IR), *mutual information score* (MI), and *answer frequency*. The *keyword overlap* feature represents the ratio of question keywords found in a passage, as used in [Cooper and Ruger 2000; Molla and Gardiner 2005; Zhao *et al*. 2005]. The *IR score* [Kwok and Deng 2006; Zheng 2002], which is provided by the passage retrieval module, is the score of the passage containing the answer. In ASQA2, the *IR score* is produced by the Lucene information retrieval engine[6]. *Density* is defined as the average distance between the answer and question keywords in a passage. There are several ways to calculate density. In this experiment, we simply adopt Lin's formula [Lin *et al*. 2005], which performed well in NTCIR-5 CLQA. The *mutual information score* is calculated by the PMI method used in [Magnini *et al*. 2001], and instead of being based on the Web, it is calculated based on the whole corpus.

The experiment results are listed in Table 3. SCO-QAT performs very well on C-C datasets, achieving 0.522 EAA for the NTCIR5-CC-D200e dataset, 0.515 for the NTCIR5-CC-T200e dataset, 0.546 for the IASL-CC-Q465 dataset, and 0.406 for the NTCIR6-CC-T150 dataset. Compared to other features, the differences are in the range 0.063~0.522 for EAA.

[6] We adopted Lucene 2.0.0, which uses Vector Space Model as the default method to calculate the IR score of a document. Detail information can be found in the Lucene API documentation: Class Similarity:http://lucene.apache.org/java/2_0_0/api/org/apache/lucene/search/Similarity.htmlClass DefaultSimilarity:
http://lucene.apache.org/java/2_0_0/api/org/apache/lucene/search/DefaultSimilarity.html

Table 3. The performance of single features. “Accuracy” is the RU-Accuracy, “MRR” is Top5 RU-Mean-Reciprocal-Rank scores, and “EAA” is the Expected Answer Accuracy. CC-ALL and EC-ALL are the respective combinations of all the CC and EC datasets.

	NTCIR5-CC-D200e			**NTCIR5-CC-T200e**		
	Accuracy	EAA	MRR	Accuracy	EAA	MRR
SCOQAT	**0.545**	**0.522**	**0.621**	**0.515**	**0.515**	**0.586**
KO	0.515	0.254	0.601	0.495	0.245	0.569
Density	0.375	0.368	0.501	0.390	0.380	0.479
Frequency	0.445	0.431	0.560	0.395	0.366	0.499
IR	0.515	0.425	0.598	0.495	0.420	0.569
MI	0.210	0.210	0.342	0.155	0.290	0.138
	IASL-CC-Q465			**NTCIR6-CC-T150**		
	Accuracy	EAA	MRR	Accuracy	EAA	MRR
SCOQAT	**0.578**	**0.546**	**0.628**	**0.413**	**0.406**	**0.495**
KO	0.568	0.247	0.618	0.367	0.130	0.476
Density	0.432	0.369	0.519	0.340	0.314	0.420
Frequency	0.413	0.406	0.486	0.340	0.343	0.431
IR	0.518	0.406	0.587	0.367	0.283	0.460
MI	0.138	0.124	0.280	0.167	0.142	0.281
	NTCIR5-EC-D200			**NTCIR5-EC-T200**		
	Accuracy	EAA	MRR	Accuracy	EAA	MRR
SCOQAT	0.250	0.240	0.349	0.185	0.187	0.265
KO	0.290	0.117	0.376	0.195	0.093	0.288
Density	0.190	0.186	0.294	0.180	0.177	0.245
Frequency	**0.300**	**0.297**	**0.394**	0.190	0.181	0.280
IR	0.295	0.262	0.385	**0.270**	**0.210**	**0.326**
MI	0.145	0.145	0.262	0.060	0.046	0.164
	NTCIR6-EC-T150					
	Accuracy	EAA	MRR			
SCOQAT	0.193	0.180	0.268			
KO	**0.220**	0.061	**0.292**			
Density	0.187	0.180	0.268			
Frequency	0.213	0.194	0.283			
IR	0.180	**0.265**	0.146			
MI	0.107	0.069	0.205			
	CC-ALL			**EC-ALL**		
	Accuracy	EAA	MRR	Accuracy	EAA	MRR
SCOQAT	**0.535**	**0.514**	**0.599**	0.211	0.204	0.296
KO	0.513	0.231	0.584	0.236	0.093	0.321
Density	0.399	0.363	0.493	0.185	0.181	0.269
Frequency	0.405	0.394	0.495	0.236	**0.227**	0.322
IR	0.491	0.424	0.538	**0.255**	0.212	**0.331**
MI	0.160	0.176	0.264	0.104	0.088	0.211

In addition to comparing single ranking features, we compared the SCO-QAT results with those of other participants in the NTCIR5 CLQA task (Table 4). As the other QA systems used combined features, this is a single- versus combined-feature comparison. In the NTCIR5 CLQA task [Sasaki *et al*. 2005], there were thirteen Chinese QA runs with an accuracy range of 0.105~0.445, and a mean of 0.315. It is impressive that ASQA2 with the single SCO-QAT feature achieved 0.515 accuracy[7], which was much better than the accuracy of ASQA1 [Lee *et al*. 2005], the best performing system in the NTCIR5 CLQA C-C subtask.

Table 4. Performance comparison of SCO-QAT (single feature) and the best systems at NTCIR5 and NTCIR6 CLQA (combined features)

Subtask	System	RU-Accuracy
NTCIR5 CC	Best Participant (ASQA1)	0.445
	ASQA2 with SCO-QAT only	0.515
NTCIR5 EC	Best Participant	0.165
	ASQA2 with SCO-QAT only	0.185
NTCIR6 CC	Best Participant (ASQA2 full version)	0.553
	ASQA2 with SCO-QAT only	0.413
NTCIR6 EC	Best Participant (ASQA2 full version)	0.340
	ASQA2 with SCO-QAT only	0.193

Although SCO-QAT still performs well on the E-C datasets, its performance is not as good as on the C-C datasets. After analyzing the failed cases of E-C QA, we found the major problem was that some translations introduced words not listed in the stop word list. For example, there were some English questions in NTCIR CLQA, such as “Who is in charge of Indonesia's cabinet in 2000?” After processing their Google translations, we identified improper keywords that were not on our stop word lists. For example, in the translation of the above question, “由誰負責的印尼內閣於 2000 年?”, we found “由” and “於”. Since SCO-QAT aggregates all co-occurrence scores, the effect of improper keywords is compounded. Although this problem could be solved by simply adding more stop words to the list, it should be noted that more new stop words may be introduced if the machine translation engine is changed. A better solution is to use the term-by-term translation approach because the stop word list can be controlled more easily.

Although *frequency* is the simplest of the shallow features, it performs surprisingly well. It even achieves the best performance on one E-C dataset (NTCIR5-EC-D200). This may be

[7] The 0.515 accuracy is based on NTCIR5-CC-T200e dataset. If based on the NTCIR5-CC-T200 dataset, the accuracy is 0.505

due to the effectiveness of the ASQA2 answer filtering module, the characteristics of the Chinese news corpus, or the way questions were created, which caused questions with high frequency answers to be selected. We cannot find any papers on the effect of applying the frequency feature only. Further investigation is, therefore, needed to explain the phenomenon.

The density feature measures the density of question terms around the answer based on the co-occurrence and distance information. Although it is widely used in QA systems, its performance is not as good as that of the IR score, which does not consider the distance information. This could be because the distance information is much noisier in QA that involves Chinese (*e.g.*, E-C and C-C).

We identified two types of errors caused by machine translations: wrong-term errors and synonym errors. Both types have a negative effect on the ranking features because the quality of the passages is often poor. The following is an example of a wrong term error. For the English question “Who is the director of the Chinese movie Crouching Tiger, Hidden Dragon?”, the word “director” was translated by Google Translate to the wrong term “新任” in “誰是新任的中國電影臥虎藏龍?”. Here, the semantics of “director” and “新任” are completely different. In cases like this, it is impossible to find good quality passages for ranking. Synonym errors occur when improper synonyms are introduced. For example, the English question “Who was Taiwan's Central Bank Governor with the longest tenure?” is translated to “誰是台灣的央行行長最長任期?” by Google. Although “行長” is the correct translation for mainland China, it is not the normal way to describe the head of a bank in Taiwan; therefore, a query with “行長” can not retrieve appropriate passages from Taiwanese news corpora (*e.g.*, CIRB40 and CIRB20).

6.4 Experiment 2 –Influence of Machine Translation Quality

To develop a cross-lingual QA system, a monolingual system is usually created first and then some modules are adjusted to meet cross-lingual requirements. There are two widely used approaches: question translation and term-by-term translation. In the question translation approach, the question is translated into the target language by machine translation. The translated question is then input to the monolingual system. In the term-by-term approach, questions are analyzed in the source language and split into several important terms, which are then translated by using a bilingual dictionary or other techniques.

Since ASQA2 adopts the question translation approach, we can control the translation quality intuitively using different machine translation engines. Noisy information introduced by a machine translation engine propagates down through the QA modules and results in wrong answers. We tested our system on two machine translation services (namely, Google

Translate and SYSTRAN[8]) to determine how the translation quality affects the answer ranking features. Table 5 shows the experimental set-up.

Table 5. Experimental Set-up for Experiment 2 – Influence of Machine Translation Quality

Independent Variables	Ranking Feature, Translation Engine
Dependent Variables	Accuracy, MRR, EAA
Controlled Variables	Passage Depth, Mono- or Cross-lingual, Answer Filter

We observe that Google's translation quality is better than that of SYSTRAN. In other words, the accuracy declines when Google Translate is replaced by SYSTRAN. The performance decrease ratio (calculated as the performance of using SYSTRAN divided by that of using Google) for each of the three E-C datasets is shown in Table 6. It seems to be difficult to predict the influence of the translation quality. If we only look at each dataset, the decrease ratio is quite unstable, ranging from 48.3% to 96.9% in terms of accuracy. However, when we consider the ratio based on all the datasets, it becomes more stable for all the ranking features. The standard deviation of the decrease in the accuracy ratio drops from more than 0.11 to 0.0655, which shows that the current datasets of NTCIR CLQA may be too small to be used with confidence in our experiments. Thus, it would be better to use all the EC datasets when comparing QA systems.

For the EC-ALL dataset, SCO-QAT yields a better performance decrease ratio in terms of accuracy and EAA, but not in terms of MRR. The Frequency feature still performs relatively well, because the frequency of an answer is less dependant on the translation quality.

[8] We used the Yahoo! BABEL FISH service, which is powered by SYSTRAN. The translations were obtained from Google and Yahoo in May 2007 and June 2007, respectively.

Table 6. Performance decrease ratio of shallow features on E-C QA when Google is replaced by SYSTRAN.

	(a) NTCIR5-EC-D200			(b) NTCIR5-EC-T200		
	Accuracy	EAA	MRR	Accuracy	EAA	MRR
SCOQAT	**80.0%**	79.2%	62.6%	59.5%	59.0%	67.4%
KO	69.0%	**83.5%**	76.2%	51.3%	58.7%	60.9%
Density	73.7%	75.3%	78.7%	61.1%	**59.8%**	68.2%
Frequency	73.3%	68.8%	77.5%	47.4%	47.4%	62.8%
IR	79.7%	80.5%	**78.9%**	35.2%	45.1%	49.3%
MI	48.3%	34.5%	66.8%	**91.7%**	72.2%	**79.0%**
Stdev.	*0.1173*	*0.1826*	*0.0697*	*0.1910*	*0.0980*	*0.0980*
	(c) NTCIR6-EC-T150			(d) EC-ALL		
	Accuracy	EAA	MRR	Accuracy	EAA	MRR
SCOQAT	82.8%	86.2%	83.1%	**74.1%**	**74.2%**	69.2%
KO	87.9%	71.9%	85.7%	68.5%	72.4%	73.6%
Density	89.3%	88.3%	85.7%	73.5%	73.3%	**77.1%**
Frequency	**96.9%**	**97.6%**	**91.3%**	71.5%	69.3%	76.2%
IR	66.7%	62.3%	71.3%	60.0%	64.3%	66.6%
MI	56.2%	51.0%	72.6%	59.6%	45.1%	71.8%
Stdev.	*0.1538*	*0.1762*	*0.0794*	*0.0655*	*0.1105*	*0.0403*

6.5 Experiment 3 –Influence of Passage Quality Introduced by Deep Passages

Passage depth, defined as the number of passages used for answer extraction and answer ranking, plays a critical role in a QA system. On the one hand, by increasing the passage depth we can obtain more relevant passages and, therefore, have a better chance of improving QA performance. On the other hand, increasing the passage depth also introduces more irrelevant passages. If a ranking feature can not handle the noise caused by deep passages, it can not benefit from additional relevant passages.

In this experiment, we increase the number of passages to evaluate the performance of shallow features when the number of irrelevant passages increases. The experimental setup is shown in Table 7.

Table 7. Experimental Set-up for Experiment 3 – Influence of Passage Quality Introduced by Deep Passages.

Independent Variables	Ranking Feature, Passage Depth, Mono-or Cross-lingual
Dependent Variables	Accuracy, MRR, EAA
Controlled Variables	Translation Engine, Answer Filter

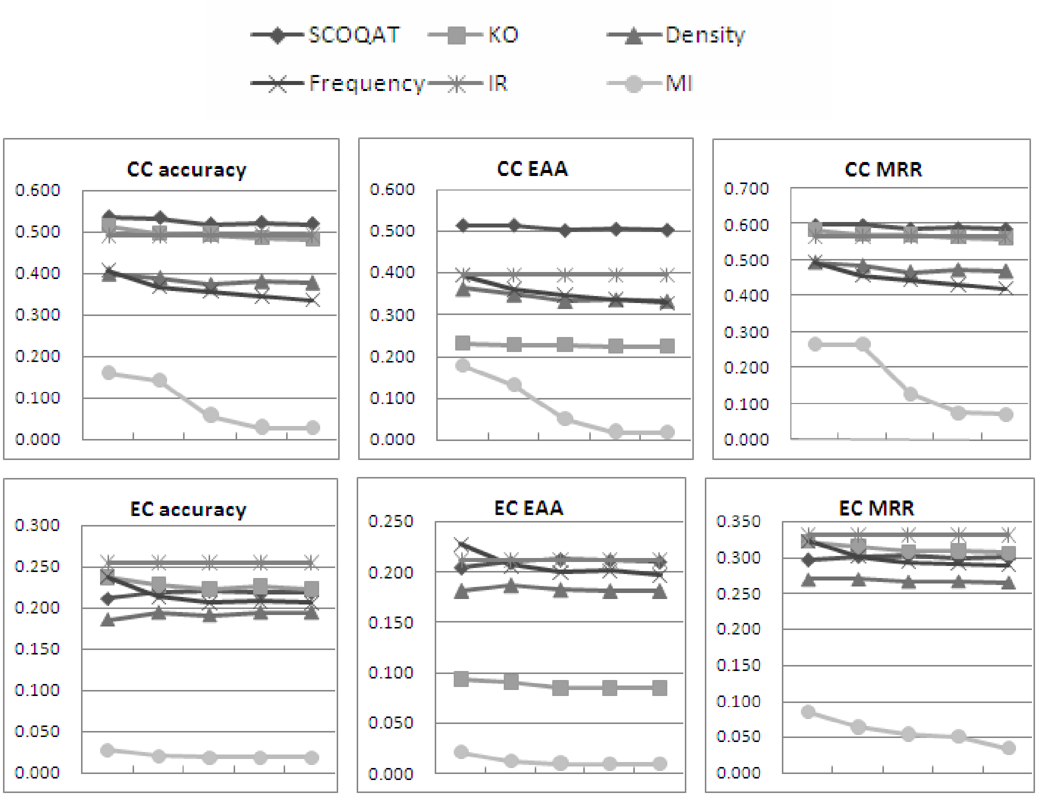

Figure 4. Single feature accuracy over 5 passage depth points (100, 200, 300, 400, 500) for all C-C and E-C datasets.

We observe the performance of all C-C and E-C datasets at five depth points between 100 and 500, as shown in Figure 4. We chose 100 as the starting depth because it is commonly adopted in QA systems as the document depth or passage depth. As expected, for both CC and EC situations, EAA declines when the passage depth increases. (The IR score ranking feature is an exception. It always remains the same because the passage IR score of an answer does not change when the passage depth increases). However, the decrease in EAA is not as high as we expected, which suggests that, with the exception of frequency and MI, shallow ranking features can handle deep passage noise.

Among the ranking features, *frequency* and *MI* are influenced by passage depth the most. In EC, while *frequency* is the best at depth 100 in terms of EAA, the latter decreases rapidly when the passage depth increases to 200, which is much more unreliable than in the CC situation. In other words, the *accuracy* feature is much more unreliable in EC. For *MI*, it not only performed worse than the other features in terms of EAA, but also decreased substantially when the depth increased. This suggests that *MI* may not be suitable for retrieved passages, although it has been applied successfully when using the Web as a corpus.

Some of the examples found confirm that the number of irrelevant passages increases when the number of passages increases. For example, when the number of passages is 100, the most frequent answer given to the Chinese question "西元 2000 年加入奧地利聯合政府的自由黨黨魁是誰？" (Who is the leader of Freedom Party joining the Austria coalition government in 2000?) is "海德" (Haider), which is correct. However, when the number increases to 200, the most frequent answer is "小澤一郎" (OZAWA Ichiro), which is incorrect. This causes *density* and the other shallow features to fail in this situation.

6.6 Experiment 4 –Influence of Answer Quality

As answer ranking is directly influenced by the answer quality, it is important to evaluate the ranking feature on answers of different quality. In this experiment, we adjusted the answer quality by changing the answer filter. The experimental set-up is detailed in Table 8.

Table 8. Experimental Set-Up for Experiment 4 – Influence of Answer Quality

Independent Variables	Ranking Feature, Mono- or Cross-lingual
Dependent Variables	Accuracy, MRR, EAA
Controlled Variables	Passage Depth, Translation Engine, Answer Filter

The Expected Answer Type filter (EAT filter) is a submodule of ASQA2 that eliminates answers deemed incompatible with the question type. For example, if the question type is Q_LOCATION_COUNTRY, only answers representing countries will be retained. It is common for QA systems to use this kind of filtering mechanism, but they differ in the granularity of the answer type system they use. With a good EAT filter, the quality of the input for the subsequent Answer Ranking module will be less noisy and easier to deal with.

By utilizing the ASQA2 answer-type system (*i.e.*, 6 coarse-grained and 62 fine-grained types), we can experiment with answer ranking features on different granularities. We built three EAT filters, namely, a DoNothing Filter, a Coarse-grained Filter[9], and a Fine-grained Filter. The DoNothing Filter does not filter out any answers; therefore, it may contain a lot of noisy information. The Coarse-grained Filter and Fine-grained Filter use coarse-grained and

[9] The Fine-grained filter was used in ASQA1 and ASQA2

fine-grained type information respectively.

The Fine-grained Filter is used in the single feature experiment described in Section 6.3. Here, we conduct the same single feature experiment with the other two noisier EAT filters. The results are shown in Table 9. As expected, the performance of every feature deteriorates with the noisy EAT filters. In the CC datasets, with the Coarse-grained Filter, SCO-QAT's EAA declines from 0.514 to 0.499 on the CC-ALL dataset, but it is still better than the other features. Even with the noisiest DoNothing Filter, SCO-QAT can still maintain a 71% decrease ratio for the CC-ALL dataset, thereby demonstrating its robustness. The calculation of decrease ratios in this section is similar to that in the "Influence of Machine Translation Quality" section. When speaking of Coarse-grained Filter, it is calculated as the performance of using Coarse-grained Filter divided by the performance of using Fine-grained Filter. When speaking of DoNothing Filter, the formula is the same except that the numerator is replaced with the performance of using DoNothing Filter.

Table 9(a). Performance and decrease ratio in CC QA when the Coarse-grained EAT filter is replaced by Fine-grained and DoNothing EAT filters.

Coarse-Grained (Decrease Ratio = Coarse-Grained / Fine-Grained)						
	(a) NTCIR5-CC-D200e			(b) NTCIR5-CC-T200e		
	Accuracy	EAA	MRR	Accuracy	EAA	MRR
SCOQAT	**0.515** (94%)	**0.492** (94%)	**0.594** (**96%**)	**0.5** (**97%**)	**0.498** (**97%**)	**0.56** (**96%**)
KO	0.475 (92%)	0.229 (90%)	0.564 (94%)	0.48 (**97%**)	0.224 (92%)	0.545 (**96%**)
Density	0.355 (**95%**)	0.35 (**95%**)	0.473 (95%)	0.345 (88%)	0.333 (88%)	0.441 (92%)
Frequency	0.41 (92%)	0.408 (**95%**)	0.524 (93%)	0.37 (94%)	0.344 (94%)	0.472 (95%)
IR	0.475 (92%)	0.392 (92%)	0.559 (94%)	0.465 (94%)	0.375 (89%)	0.539 (95%)
MI	0.035 (17%)	0.031 (15%)	0.089 (26%)	0.04 (26%)	0.034 (12%)	0.104 (75%)
	(c) IASL-CC-Q465			(d) NTCIR6-CC-T150		
	Accuracy	EAA	MRR	Accuracy	EAA	MRR
SCOQAT	**0.568** (**98%**)	**0.536** (**98%**)	**0.619** (**99%**)	**0.407** (98%)	**0.398** (98%)	**0.486** (98%)
KO	0.551 (97%)	0.232 (94%)	0.604 (98%)	0.367 (**100%**)	0.123 (95%)	0.468 (98%)
Density	0.406 (94%)	0.337 (91%)	0.498 (96%)	0.327 (96%)	0.301 (96%)	0.405 (97%)
Frequency	0.394 (95%)	0.385 (95%)	0.468 (96%)	0.34 (100%)	0.339 (**99%**)	0.43 (**100%**)
IR	0.508 (**98%**)	0.39 (96%)	0.576 (98%)	0.367 (100%)	0.269 (95%)	0.45 (98%)
MI	0.03 (22%)	0.02 (16%)	0.095 (34%)	0.06 (36%)	0.032 (23%)	0.124 (44%)
	(e) CC-ALL					
	Accuracy	EAA	MRR			
SCOQAT	**0.52** (**97%**)	**0.499** (**97%**)	**0.583** (**97%**)			
KO	0.495 (96%)	0.214 (93%)	0.564 (**97%**)			
Density	0.372 (93%)	0.333 (92%)	0.468 (95%)			
Frequency	0.384 (95%)	0.374 (95%)	0.474 (96%)			
IR	0.472 (96%)	0.369 (94%)	0.547 (**97%**)			
MI	0.037 (24%)	0.027 (16%)	0.1 (42%)			

Table 9 also shows the performance decrease ratio caused by inefficient EAT filters. It is calculated by dividing the performance score of a noisy EAT filter by that of the standard Fine-grained Filter. From this perspective, SCO-QAT is still the best CC feature, achieving 97% and 71% EAA decrease ratio with the Coarse-Grained Filter and DoNothing EAT filter, respectively.

Table 9(b). Performance and decrease ratio in CC QA when the Coarse-grained EAT filter is replaced by the Fine-grained and DoNothing EAT filters.

DoNothing (Decrease Ratio = DoNothing / Fine-Grained)						
	(f) NTCIR5-CC-D200e			(g) NTCIR5-CC-T200e		
	Accuracy	EAA	MRR	Accuracy	EAA	MRR
SCOQAT	**0.355** (65%)	**0.339** (65%)	**0.463** (**74%**)	**0.345** (**67%**)	**0.341** (**66%**)	**0.442** (**76%**)
KO	0.345 (**67%**)	0.082 (32%)	0.452 (75%)	0.315 (64%)	0.068 (28%)	0.414 (73%)
Density	0.16 (43%)	0.14 (38%)	0.16 (32%)	0.185 (47%)	0.179 (47%)	0.275 (57%)
Frequency	0.3 (**67%**)	0.285 (**66%**)	0.395 (71%)	0.23 (58%)	0.22 (60%)	0.331 (66%)
IR	0.32 (62%)	0.135 (32%)	0.43 (72%)	0.335 (68%)	0.152 (36%)	0.428 (75%)
MI	0.02 (10%)	0.018 (9%)	0.108 (32%)	0.015 (10%)	0.005 (2%)	0.036 (26%)
	(h) IASL-CC-Q465			(i) NTCIR6-CC-T150		
	Accuracy	EAA	MRR	Accuracy	EAA	MRR
SCOQAT	**0.428** (74%)	**0.406** (**74%**)	**0.52** (**83%**)	**0.293** (**71%**)	**0.295** (**73%**)	**0.374** (**76%**)
KO	0.426 (**75%**)	0.061 (25%)	0.513 (**83%**)	0.24 (65%)	0.034 (26%)	0.333 (70%)
Density	0.254 (59%)	0.179 (48%)	0.343 (66%)	0.153 (45%)	0.131 (42%)	0.246 (59%)
Frequency	0.288 (70%)	0.285 (70%)	0.356 (73%)	0.22 (65%)	0.223 (65%)	0.304 (70%)
IR	0.376 (73%)	0.211 (52%)	0.473 (80%)	0.24 (65%)	0.124 (44%)	0.331 (72%)
MI	0.013 (9%)	0.003 (2%)	0.04 (14%)	0.007 (4%)	0.001 (1%)	0.027 (10%)
	(j) CC-ALL					
	Accuracy	EAA	MRR			
SCOQAT	**0.377** (**70%**)	**0.364** (**71%**)	**0.472** (**79%**)			
KO	0.361 (**70%**)	0.063 (27%)	0.455 (78%)			
Density	0.207 (51%)	0.164 (45%)	0.279 (57%)			
Frequency	0.269 (66%)	0.263 (67%)	0.351 (71%)			
IR	0.337 (69%)	0.171 (44%)	0.435 (77%)			
MI	0.014 (9%)	0.006 (3%)	0.051 (19%)			

The decline in some features is caused by too many answers being collocated in the same passage. Without a proper EAT filter, a passage could contain the correct answer and other answers; or, at worst, contain several answers, none of which are compatible with the given question. For example, the first returned passage for the Chinese question “請問西元2000年7月美方派何人前往北京對TMD以及其他全球戰略佈局與中方展開對話？” (Who is the delegate of United States visiting Beijing to negotiate the TMD issue in July, 2000?) does not contain any answers related to the PERSON type. Without a proper filter, wrong answers in the top-ranked passages would be sent to the answer ranking module. As a result, the IR score would not help us differentiate between the correct answer and incorrect ones.

Note that the decline in EC's performance is substantial when the DoNothing filter is applied. In the CC case, the decline in EAA for the SCO-QAT feature is 71%; however, in the EC case, it drops to 14%. This suggests that, in EC, information about the answer type is important, since it is more reliable than the shallow ranking features under noise introduced by translation.

Table 10. Performance and decrease ratio in EC QA when the Coarse-grained EAT filter is replaced by the Fine-grained and DoNothing EAT filters.

Coarse-Grained (Coarse-Grained / Fine-Grained)						
	(a) NTCIR5-EC-D200			(b) NTCIR5-EC-T200		
	Accuracy	EAA	MRR	Accuracy	EAA	MRR
SCOQAT	0.2 (80%)	0.1947 (81%)	0.3019 (86%)	0.17 (**91%**)	0.1702 (**91%**)	0.2431 (**91%**)
KO	**0.255** (**87%**)	0.102 (**87%**)	0.334 (**88%**)	0.155 (79%)	0.07 (75%)	0.2499 (86%)
Density	0.16 (84%)	0.1537 (82%)	0.2517 (85%)	0.15 (83%)	0.1442 (81%)	0.2183 (88%)
Frequency	**0.255** (85%)	**0.2559** (86%)	**0.3486** (**88%**)	0.16 (84%)	0.1608 (88%)	0.2509 (89%)
IR	0.25 (84%)	0.2262 (86%)	0.3359 (87%)	**0.23** (85%)	**0.1826** (87%)	**0.2966** (90%)
MI	0.02 (13%)	0.0175 (12%)	0.0944 (36%)	0.015 (25%)	0.0106 (23%)	0.0655 (39%)
	(c) NTCIR6-EC-T150			(d) EC-ALL		
	Accuracy	EAA	MRR	Accuracy	EAA	MRR
SCOQAT	0.1867 (96%)	0.1711 (95%)	0.2586 (96%)	0.1855 (**87%**)	0.1794 (87%)	0.2687 (**90%**)
KO	0.1867 (84%)	0.0591 (97%)	**0.2702** (92%)	0.2 (84%)	0.0787 (84%)	0.286 (89%)
Density	0.18 (96%)	0.1766 (98%)	0.2559 (95%)	0.1618 (**87%**)	0.1565 (86%)	0.2407 (89%)
Frequency	**0.1933** (90%)	**0.1769** (91%)	0.268 (94%)	0.2036 (86%)	**0.1998** (87%)	0.2911 (**90%**)
IR	0.18 (**100%**)	0.1449 (**99%**)	0.2598 (**98%**)	**0.2236** (**87%**)	0.1882 (**88%**)	**0.3009** (**90%**)
MI	0.0533 (49%)	0.0391 (56%)	0.1108 (53%)	0.0273 (26%)	0.0209 (23%)	0.0884 (41%)
DoNothing (DoNothing / Fine-Grained)						
	(e) NTCIR5-EC-D200			(f) NTCIR5-EC-T200		
	Accuracy	EAA	MRR	Accuracy	EAA	MRR
SCOQAT	**0.02** (**8%**)	0.0226 (9%)	0.1254 (36%)	**0.035** (**19%**)	0.035 (19%)	0.1206 (**46%**)
KO	**0.02** (7%)	**0.0232** (20%)	0.1385 (**37%**)	0.015 (8%)	0.02 (**21%**)	0.1207 (42%)
Density	0.015 (**8%**)	0.019 (10%)	0.1013 (35%)	0.02 (11%)	0.0225 (13%)	0.0934 (38%)
Frequency	**0.02** (7%)	0.0163 (5%)	0.1365 (35%)	0.01 (5%)	0.01 (6%)	0.1124 (40%)
IR	**0.02** (7%)	0.067 (**26%**)	**0.1397** (36%)	0.02 (7%)	**0.0357** (17%)	**0.1278** (39%)
MI	0 (0%)	0.0004 (0%)	0.0184 (7%)	0.005 (8%)	0.0003 (1%)	0.0199 (12%)
	(g) NTCIR6-EC-T150			(h) EC-ALL		
	Accuracy	EAA	MRR	Accuracy	EAA	MRR
SCOQAT	**0.0267** (14%)	0.0267 (15%)	0.1086 (**41%**)	**0.0273** (**13%**)	0.0282 (14%)	0.1191 (**40%**)
KO	0.02 (9%)	0.0136 (**22%**)	**0.1102** (38%)	0.0182 (8%)	**0.0194** (**21%**)	0.1243 (39%)
Density	0.02 (11%)	0.0184 (10%)	0.1061 (40%)	0.0182 (10%)	0.0201 (11%)	0.0997 (37%)
Frequency	0.02 (9%)	0.02 (10%)	0.1043 (37%)	0.0164 (7%)	0.015 (7%)	0.119 (37%)
IR	0.0133 (7%)	**0.0294** (20%)	0.1 (38%)	0.0182 (7%)	**0.0453** (**21%**)	**0.1245** (38%)
MI	**0.0267** (**25%**)	0.0041 (6%)	0.0464 (23%)	0.0091 (9%)	0.0013 (2%)	0.0266 (13%)

7. Conclusion

Sometimes, the resources needed to apply deep answer ranking approaches in a language are not available or the resource quality is not good enough. Hence, we conducted this research to help QA system designers choose shallow ranking features. We experimented on six shallow ranking features (SCO-QAT, keyword overlap, density, IR score, mutual information score, and answer frequency) under various types of noise caused by different QA modules in mono-lingual and cross-lingual situations.

We also proposed a novel answer ranking feature called SCO-QAT, which does not require extra knowledge or sophisticated tools. It is, therefore, easy to implement in QA systems and may be used on various languages. In this pilot study, when the ASQA2 system only used the SCO-QAT ranking feature, it outperformed all the systems in NTCIR5 CLQA. For example, on the NTCIR5-CC-T200e QA dataset, we achieved 0.515 RU-Accuracy with the SCO-QAT feature only. Even the E-C version also achieved a 0.05 improvement over the best system. SCO-QAT also performed well in NTCIR6 CLQA, where the host system, ASQA2, achieved the best performance in the C-C subtask and the E-C subtask.

To understand SCO-QAT better and to gain a deeper insight into shallow answer ranking features, we tested answer ranking features in various scenarios. We found that, although SCO-QAT performed very well in C-C QA, frequency seems the best choice for ranking in E-C QA in terms of EAA. However, the decrease in translation quality has a marked effect on the frequency of EAA, as shown by the fact that the EAA decrease ratio is 69.3%. In the same situation, SCO-QAT maintained a 74.2% EAA decrease ratio which was the best among the shallow ranking features. We also found that the noise introduced by passage depth does not impact much on ranking performance. This suggests that, if a long processing time is allowed, QA based on deep passages is a possible way to improve the performance when shallow features are used. In addition, answer-type-based filtering plays an important role, especially for E-C. When an extremely bad filter was used, the EAA decrease ratio in E-C for shallow ranking features was only 2%~21%, which shows a proper answer filter with fined-grained NER is critical to the success of an E-C system.

In our future research on shallow ranking features, we will address the following issues. We will introduce a question term weighting scheme for SCO-QAT; use a taxonomy or ontology to alleviate the synonym problem that arises when counting co-occurrences of answers and question terms; experiment with shallow features on a Web corpus; utilize more syntactic information to make co-occurrence information more reliable; and test shallow features on other languages.

Acknowledgments

This research was supported in part by the National Science Council of Taiwan under Center of Excellence Grant NSC 95-2752-E-001-001-PAE, the Research Center for Humanities and Social Sciences, Academia Sinica, and Thematic program of Academia Sinica under Grant AS 95ASIA02. We would like to thank the Chinese Knowledge and Information Processing Group (CKIP) in Academia Sinica for providing us with AutoTag for Chinese word segmentation.

REFERENCES

Clarke, C.L.A., G. Cormack, G. Kemkes, M. Laszlo, T. Lynam, E. Terra, and P. Tilker, "Statistical Selection of Exact Answers (MultiText Experiments for TREC 2002)," in *Proc. of TREC, 2002*, pp. 823-831.

Clarke, C.L.A., G.V. Cormack, and T.R. Lynam, "Exploiting redundancy in question answering," in *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, 2001, pp. 358-365.

Cooper, R.J. and S.M. Ruger, "A Simple Question Answering System," in *Proc. of TREC*, 2000.

Cui, H., R. Sun, K. Li, M.Y. Kan, and T.S. Chua, "Question answering passage retrieval using dependency relations," in *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, 2005, pp. 400-407.

Geleijnse, G. and J. Korst, "Learning Effective Surface Text Patterns for Information Extraction," in *Proceedings of the EACL 2006 workshop on Adaptive Text Extraction and Mining (ATEM 2006)*, 2006, pp. 1-8.

Gillard, L., L. Sitbon, E. Blaudez, P. Bellot, and M. El-B`eze, "The LIA at QA@CLEF-2006," in *CLEF*, 2006.

Harabagiu, S., D. Moldovan, C. Clark, M. Bowden, A. Hickl, and P. Wang, "Employing Two Question Answering Systems in TREC 2005," in *Proceedings of the Fourteenth Text REtrieval Conference*, 2005.

Kwok, K.-L. and P. Deng, P., "Chinese Question-Answering:Comparing Monolingual with English-Chinese Cross-Lingual Results," in *Asia Information Retrieval Symposium*, 2006, pp. 244-257.

Lee, C.-W., M.-Y. Day, C.-L. Sung, Y.-H. Lee, T-J. Jiang, C-W. Wu, C-W. Shih, Y-R. Chen, .and W.-L. Hsu, "Chinese-Chinese and English-Chinese Question Answering with ASQA at NTCIR-6 CLQA," in *Proceedings of NTCIR-6 Workshop*, 2007, pp. 175-181.

Lee, C.W., C.W. Shih, M.Y. Day, T.H. Tsai, T.J. Jiang, C.W. Wu, C.L. Sung, Y.R. Chen, S.H. Wu, and W.L. Hsu, "ASQA: Academia Sinica Question Answering System for NTCIR-5 CLQA," in *Proceedings of NTCIR-5 Workshop Meeting* , 2005, Tokyo, Japan.

Lin, F., H. Shima, M. Wang, and T. Mitamura, "CMU JAVELIN System for NTCIR5 CLQA1," in *Proceedings of the 5th NTCIR Workshop*, 2005.

Lin, J., "Evaluation of resources for question answering evaluation," in *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 392-399.

Lin, S.-J., M.-S. Shia, K.-H. Lin, J.-H. Lin, S. Yu, and W.-H. Lu, "Improving answer ranking using cohesion between answer and keywords," in *NTCIR Workshop*, 2005.

Magnini, B., M. Negri, R. Prevete, and H. Tanev, "Is it the right answer?: exploiting web redundancy for Answer Validation," in *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, 2001, pp. 425-432.

Molla, D. and M. Gardiner, M., "AnswerFinder — Question Answering by Combining Lexical, Syntactic and Semantic Information," in *Australasian Language Technology Workshop (ALTW) 2004*, Sydney, Australia, pp. 9-16.

Ravichandran, D. and E. Hovy, "Learning Surface Text Patterns for a Question Answering System,"in *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pp. 41-47.

Roussinov, D., J. Robles, and Y. Ding, "Experiments with Web QA System and TREC2004 Questions," in *the proceedings of TREC conference*, November, 2004, pp. 16-19.

Sacaleanu, B. and G. Neumann, "DFKI-LT at the CLEF 2006 Multiple Language Question Answering Track," in *CLEF, 2006*.

Sasaki, Y., H.H. Chen, K. Chen, and C.J. Lin, "Overview of the NTCIR-5 Cross-Lingual Question Answering Task (CLQA1)," in P*roceedings of the Fifth NTCIR Workshop Meeting*, pp. 6-9.

Sasaki, Y., C.-J. Lin, K-H. Chen, and H.-H. Chen, "Overview of the NTCIR-6 Cross-Lingual Question Answering (CLQA) Task," in *Proceedings of NTCIR-6 Workshop*, 2007, Tokyo, Japan.

Shen, D., G. Saarbruecken, and D. Klakow, "Exploring Correlation of Dependency Relation Paths for Answer Extraction," in *Proceedings of ACL 2006*, 2006, Sydney, Australia, pp. 889-896.

Soubbotin, M.M. and S.M. Soubbotin, "Patterns of Potential Answer Expressions as Clues to the Right Answers," in *Proceedings of the Tenth Text REtrieval Conference (TREC 2001*), 2001, Gaithersburg, MD, pp. 134-143.

Tom´as, D., J.e.L. Vicedo, E. Bisbal, and L. Moreno, "Experiments with LSA for Passage Re-Ranking in Question Answering," in *CLEF*, 2005.

Zhao, Y., Z.M. Xu, Y. Guan, and P. Li, "Insun05QA on QA track of TREC2005," in *TREC*, 2005, Gaithersburg, MD.

Zheng, Z., "AnswerBus Question Answering System," in *Proceeding of Human Language Technology Conference*, 2002, San Diego, CA, pp. 24-27.