NTCIR-6 Monolingual Chinese and English-Chinese Cross-Lingual Question Answering Experiments using PIRCS

Kui-Lam Kwok, Peter Deng and Norbert Dinstl

Computer Science Department, Queens College, City University of New York, Flushing, NY 11367, USA
kwok@ir.cs.qc.edu, peterqc@yahoo.com, emc21@earthlink.net

Abstract

We continue to employ a minimal approach for our Chinese QA work that requires only a COTS entity extraction software and other home-built tools. In monolingual Chinese QA, questions are classified based on cue-word and meta-keyword usage patterns. Retrieval is done using sentence units, and indexing is based on bigrams and characters. Entities extracted from retrieved sentences form a pool of answer candidates which are ranked using five evidence factors. Our best monolingual result shows that when only Top1 answers are considered, 63 questions out of 150 are answered correctly with sentence support, giving an accuracy and MRR of 0.42. When unsupported answers are included, these values improve to 0.4467. English-Chinese CLQA starts with English question classification also based on an approach similar to Chinese. Three paths of translation render the question into Chinese strings. Otherwise procedures of retrieval and answer ranking remain the same as monolingual but with different parameter values. Our best run returns corresponding Top1 values as .2533 and .28 (unsupported). These are about 60% of monolingual effectiveness within our system. Effectiveness with Top2-5 answers as well as the influence of different evidence factors are also reported.

Keywords: English-Chinese Cross-lingual Question Answering; Monolingual Chinese Question Answering; candidate answer ranking; web-assisted entity translation.

1 Introduction

We follow a familiar approach to factoid QA depicted in Fig.1: given a question, it was first classified as to its expected answer category, and preprocessed to form an IR query to retrieve document sentences from the target collection. From these sentences, entities were extracted to form a candidate pool, and a scoring procedure was used to rank the candidates for output as answer(s). For English-Chinese CLQA, an extra step of translation was employed to render the original English question into a Chinese query.

We continue to employ our simple methodology from last year for the NTCIR-6 QA tasks. The main changes for this year include: question classification is done using pattern matching with predefined templates; answer ranking parameter values are improved based on results from NTCIR-5; identification of questions involving artifacts and their extraction from sentences are added. Both monolingual C-C and bilingual E-C QA experiments are completed for internal comparison.

Section 2 describes our Chinese monolingual QA methods and results, and Section 3 describes our E-C experiments and comparison with monolingual results. Section 4 has some additional experiments, and Section 5 has our conclusions.

2 Chinese Monolingual QA

QA has been investigated for many years, but it still remains an important topic and is one of the main tasks for the three well-known experimental forums for IR: TREC [1], CLEF [2] and NTCIR [3]. Chinese QA is a complicated task that can involve many different tools such as: word segmenter, POS tagger, parser, classifier, IR engine, entity extractor, ontology, MT software for CLQA, etc. Moreover, these tools are often statistical in nature and need large amounts of good-quality training data for them to be effective [e.g. 4, 5]. Thus, the threshold of entry to this task is high. In particular, Chinese QA is being promoted fairly recently, only having been initiated as a blind experiment in NTCIR-5 [6]. There, we introduced a minimal approach to QA that is prescriptive in nature and does not rely on the availability of training data or past results. Moreover, we cut the tools needed to a minimum so that software within our laboratory is sufficient for completing the investigation [7]. In NTCIR-6, we continue to refine our method.

2.1 Classification of Chinese Questions

One of the most important steps in QA is question
Fig. 1 Flowchart for C-C QA & E-C CLQA

analysis which tries to understand what a question wants. In NTCIR-6 [8], the same nine answer categories are used as in NTCIR-5, namely, ‘person, organization, location, date, time, money, percentage, numeral, artifact’. ‘numeral’ is any numeric entity other than money and percent. We denote a question’s class, CQ, by one of these answer categories. This information CQ, if correct, would be invaluable for a class CQ, by one of these answer categories. This than money and percent. We denote a question’s

2.2 Question Preprocessing, Indexing and Retrieval

After classification, the next step is to obtain fragments of document texts that may have high probability of containing answers to a Chinese question QC. Fig.1. As before [7], we employ the following procedures: a) each QC is processed by the Identifinder module (see Sec.2.3) to extract entities. A query qC for retrieval is then formed from QC via stop-word removal, Porter stemming, and double weighting of the extracted entities; b) documents of the target collection are segmented into sentences for retrieval and ranking against a query qC to provide tighter context for answer extraction later; c) retrieval is done using our PIRCS system with bigram and unigram indexing of query and document sentence texts. No pseudo-relevance feedback is used.

2.3 Extraction of Answer Candidates

After retrieval, the top-ranked sentences are assumed relevant to the query and have high probability of containing an answer to the corresponding question. Entities are extracted from these sentences using the BBN’s IdentiFinder software [9] and are potential answers ai. The software also assigns each entity a tag(ai) with value from one of the seven categories that are the same as the nine answer categories except for ‘numeral’ and ‘artifact’. We employ our own routine to detect numeric entities in sentences. For ‘artifact’ type, all answer identification.

2.4 Ranking of Answer Candidates

We continue to employ our simple method of candidate ranking based on five sources of evidence used before [7,10]. We assume that each candidate ai and its associated sentence Si in the candidate pool has a probability P(ai|Si|Q) of being an answer and support to question Q. The most likely candidate is the one that attains the largest probability, i.e. a = argmaxi P(ai|Si|Q) = argmaxi Σj P(ai|Si|Q) * P(Si|Q), where Σj sums over all sentences having ai. We employ intuitive estimation of factors that may proportionately reflect the values of these probabilities. P(ai|Si|Q) captures the probability of an answer in a retrieved sentence for Q; the influencing factors may include Vc: category agreement between
CQ and tag(a);

Vp: existence-in-question filtering;
and Vp: proximity of ai to question substrings in Si.
P(Sj|Q) is related to how good Si is a support to Q and the influencing factor is taken as Vp: the similarity between Q and Sj. Finally, the sum may be related to Vp: a candidate’s frequency in the retrieval list. In addition, a retrieval depth (d) is set to limit the number of retrieved sentences to be considered for extraction. Some parameter, formulae are changed from those used before to reflect training from last year’s known data. These are discussed below:

(a) Categorical Evidence: Vc
This measures the agreement between the expected answer category of the question CQ and a candidate’s entity type tag(ai). Vc is given five levels (200, 50, 20, 2, 1) corresponding to exact category agreement (CQ = tag(ai) except artifact), both within the major categories (person, location, organization), CQ is unknown/artifact and tag(ai) is artifact, CQ is unknown but tag(ai) is one of the major categories, and otherwise. The second level accounts for extractor behavior that sometimes mix up the major categories, the third level reflects the uncertainty of whether CQ = unknown is the same artifact type as in a document, while the fourth level assumes some prior preference for the major categories for those questions that failed to be classified.

(b) In-Question Evidence: Vw
It is a fact that questions seldom contain an answer string explicitly in their wordings. This leads to a binary value for Vw =0 or 1 depending if an answer candidate appears or does not appear in Q.

(c) Proximity Evidence: Vp
In our PIRCS retrieval system, a sentence is retrieved because of bigram or character overlaps (and their weights) with the query. If an entity is also identified in the sentence, we assume that the closer the entity is to the overlapped substrings, the higher the probability that the entity is an answer. For each entity candidate, a preceding proximity score Vp-pre and a succeeding score Vp-suc are accumulated. The pseudo-code for evaluating Vp-pre score follows:

Let c ∈ {a Chinese character, a numeric sequence, or an English word} and Vp-pre = 0;
//c starts preceding & adjacent to candidate entity for (c preceding a candidate) {
    score = 0;
    while (c == any_substring(qC)) {
        score += f(match-length) / g(distance-from-candidate);
    } c = previous_c || c;
    Vp-pre += score;
    c = previous_c (no match) or character_previous_to_match;
}

A long sequence of character/word match is given higher weight because of its length. This weight is also a function of the distance of the sequence from a candidate. We used f() = match-length and g() = log(1-distance-from-candidate) for monolingual QA.

A similar procedure for evaluating Vp-suc is done for coverings appearing after a candidate. The final score for proximity is:

Vp = 1 + αp * (Vp-pre + Vp-suc), with αp = 0.25.

As discussed before, we aim to keep our approach simple by matching substrings without need for word segmentation, and to work with returned sentences only without need for collection information.

(d) Sentence Similarity Evidence: Vs
If a sentence has high probability of relevance to qC, its entity candidates may be more likely to be answers. Our retrieval system provides a retrieval status value (RSV) for each sentence that reflects this probability of relevance through collection and sentence statistics of the bi-/1-grams. The proximity score in (c) also has some of this notion accounted through scoring the coverings between sentence and query substrings. For Chinese QA experiments, we have used the following function Vf:

Vs = 1 + αs * (m5*log(1+i))/log(1+m5)/h(rank)

Here, m5 counts overlaps of i characters between a sentence (of length Ls) and qC. Larger overlaps are accumulated into m5. Only the sentence retrieval rank is used with h() = rankk, and αs = 1.

(e) Candidate Frequency Evidence: Vf
Each answer candidate appears in different retrieved sentences with total occurrence frequency f. We assume that the more often a candidate occurs, the more likely it is a correct answer based on repeated confirmation. This is independent of proximity or similarity. We employ the following Vf score to capture this information:

Vf = 1 + αf * log(f), with αf = 1/3.

There are uncertainties in every step of our procedure. Question classification may not be accurate. The IR output ranking is approximate, and entity extraction and tagging can be unreliable. These procedures determine our candidate pool. The functions and assumptions used in the evidence factors for candidate ranking may also be erroneous. Combination of the five evidence sources by multiplication (V = Vc*Vw*Vp*Vs*Vf) is found to be best for final ranking of candidates than any other subsets. All evidence factors do not involve segmentation or syntactic analysis for estimation.

2.5 C-C Monolingual QA Results
Table 1 shows summary results of our three Chinese monolingual QA official runs labeled as: pircs-C-C-01 to -03 (rows with Top1 and Right[1],

---

Proceedings of NTCIR-6 Workshop Meeting, May 15-18, 2007, Tokyo, Japan
Right&U[1]) and three unofficial runs pircs-C-C-07-09 (other rows). They employ the same parameter values for the evidence factors, differing in the retrieval depth and another consideration ‘mpn’. ‘mpn’ (map numbers) means numeric mapping from ASCII numbers to Chinese characters in queries. We noticed that often in QA topics, numbers (such as year 1998) are expressed in ASCII, while such data in Chinese documents would more often be in characters (like 一九九八). After conversion, both are put back into the topic consecutively. We employed this for two of the submissions only. Thus, below each RunID in Table 1, a parenthesis entry (such as (4,mpn)) means a retrieval depth d = 4 sentences with numeric mapping. We emphasized on low d (4, 8) because NTCIR-5 results showed that dependence on d is bimodal and the low peak may be better. From the top half of Table 1, it is seen that for NTCIR-6 d=25 actually works much better and gives 63 correct and supported answers out of 150 topics at Top1 position, leading to an accuracy and mean reciprocal rank (MRR) of .42. For C-C-01 or -03 (d=4 or 8, mpn) runs, they return only 55 correct. The latter do not bring sufficient correct answers for extraction. The bottom half of Table 1 shows results when right but unsupported answers are also included as correct. The improvement is a few more correct topics. Top1 accuracy for C-C-02 becomes .4467.

If ‘mpn’ were used for the C-C-02 run (not shown in Table 1), it would give 64 correct (compared to 63). The extra correct answer comes from Question 10 (海基会 1998 年时的理事长是谁?) which has the ASCII year ‘1998’. One of the retrieved sentences is udn_xxx_19981026_0183 which matches the converted ‘一九九八’. Without ‘mpn’ mapping, this sentence has lower score and its correct answer (臺灣省政府) would not be ranked first. Mapping ASCII numbers to characters may enhance extraction or potentially improve retrieval results.

Using only Top1 answers impose a severe test on a QA system. A more relax evaluation is to additionally include results returned for Top2-5 positions. These unofficial results are labeled C-C-u-07 to -09 corresponding to C-C-01 to -03, and are shown in the other rows of Table 1. They may be useful for applications that need better recall to trade off precision. For our best run C-C-u-08, accounting for Top2-5 positions adds 10, 13, 1, 1 new right and supported answers sequentially. Top5 effectiveness increases to .5867 and MRR to .4852, meaning that 88 of the 150 questions have correct supported answers within the top 5 candidates, and one finds them at slightly above position 2 on average. The table also shows that for our system it may be best to consider up to Top 3 since the 4th-5th positions return few good answers. For Top3, effectiveness is .5733 for supported answers.

The lower half of Table 1 shows results when right but unsupported answers are also considered correct. C-C-u-08 run returns 94 correct answers with a Top5+U effectiveness and MRR+U of .6333 and .5199 respectively.

![Image](image_url)

If 'mpn' were used for the C-C-02 run (not shown in Table 1), it would give 64 correct (compared to 63). The extra correct answer comes from Question 10 (海基会 1998 年时的理事长是谁?) which has the ASCII year ‘1998’. One of the retrieved sentences is udn_xxx_19981026_0183 which matches the converted ‘一九九八’. Without ‘mpn’ mapping, this sentence has lower score and its correct answer (臺灣省政府) would not be ranked first. Mapping ASCII numbers to characters may enhance extraction or potentially improve retrieval results.

Using only Top1 answers impose a severe test on a QA system. A more relax evaluation is to additionally include results returned for Top2-5 positions. These unofficial results are labeled C-C-u-07 to -09 corresponding to C-C-01 to -03, and are shown in the other rows of Table 1. They may be useful for applications that need better recall to trade off precision. For our best run C-C-u-08, accounting for Top2-5 positions adds 10, 13, 1, 1 new right and supported answers sequentially. Top5 effectiveness increases to .5867 and MRR to .4852, meaning that 88 of the 150 questions have correct supported answers within the top 5 candidates, and one finds them at slightly above position 2 on average. The table also shows that for our system it may be best to consider up to Top 3 since the 4th-5th positions return few good answers. For Top3, effectiveness is .5733 for supported answers.

The lower half of Table 1 shows results when right but unsupported answers are also considered correct. C-C-u-08 run returns 94 correct answers with a Top5+U effectiveness and MRR+U of .6333 and .5199 respectively.

![Image](image_url)

If 'mpn' were used for the C-C-02 run (not shown in Table 1), it would give 64 correct (compared to 63). The extra correct answer comes from Question 10 (海基会 1998 年时的理事长是谁?) which has the ASCII year ‘1998’. One of the retrieved sentences is udn_xxx_19981026_0183 which matches the converted ‘一九九八’. Without ‘mpn’ mapping, this sentence has lower score and its correct answer (臺灣省政府) would not be ranked first. Mapping ASCII numbers to characters may enhance extraction or potentially improve retrieval results.

Using only Top1 answers impose a severe test on a QA system. A more relax evaluation is to additionally include results returned for Top2-5 positions. These unofficial results are labeled C-C-u-07 to -09 corresponding to C-C-01 to -03, and are shown in the other rows of Table 1. They may be useful for applications that need better recall to trade off precision. For our best run C-C-u-08, accounting for Top2-5 positions adds 10, 13, 1, 1 new right and supported answers sequentially. Top5 effectiveness increases to .5867 and MRR to .4852, meaning that 88 of the 150 questions have correct supported answers within the top 5 candidates, and one finds them at slightly above position 2 on average. The table also shows that for our system it may be best to consider up to Top 3 since the 4th-5th positions return few good answers. For Top3, effectiveness is .5733 for supported answers.

The lower half of Table 1 shows results when right but unsupported answers are also considered correct. C-C-u-08 run returns 94 correct answers with a Top5+U effectiveness and MRR+U of .6333 and .5199 respectively.

![Image](image_url)
previously [7] is also evident in these curves, except that they appear at around \( d=6 \) and \( d=28 \), different from before. Unlike NTCIR-5 results, the peak at \( d=28 \) is much higher. Setting the retrieval depth to this higher value appears to give more stable results, and is more preferable.

Table 2 shows how our system performs with respect to different question categories. At Top1 position, it returns reasonable results for person (57% correct), location (50%), date (46%) and artifact (43%) questions, fair results for organization (31%) and money (25%), but failed for ‘numex’, percent and time (0%). Our question classification for ‘numex’ and ‘time’ shows that all the 11 ‘numex’ and 2 ‘time’ questions are good. This implies that it is a retrieval or extraction problem. ‘Percent’ category however was classified correct only 1 out of 4 questions. Moreover, 4 questions on ‘money’ were wrongly classified as ‘numex’ which may explain the low performance for these categories. It was surprising that the simple strategy for ‘artifact’ detection from documents can lead to 43% accuracy.

<table>
<thead>
<tr>
<th>Query Type</th>
<th>#Q</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>1-5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Artifact</td>
<td>7</td>
<td>3</td>
<td>43</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Date</td>
<td>39</td>
<td>18</td>
<td>46</td>
<td>2</td>
<td>6</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Location</td>
<td>16</td>
<td>8</td>
<td>50</td>
<td>1</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Money</td>
<td>8</td>
<td>2</td>
<td>25</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Numex</td>
<td>11</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Organization</td>
<td>16</td>
<td>5</td>
<td>31</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Percent</td>
<td>4</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Person</td>
<td>47</td>
<td>27</td>
<td>57</td>
<td>2</td>
<td>4</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Time</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

Table 2: Chinese QA Results with Sentence Support - Category Breakdown

3 English-Chinese CLQA

As shown in Fig.1, our CLQA approach differs from monolingual only in question analysis and translation. A pattern-based program (similar to Chinese) is used to classify English questions. Here cue words include ‘Who, When, Where, What, Which, How, ..’ and meta-keywords are those that can indicate answer types such as: Which ‘president’. How ‘high’, Which ‘county’, etc [11]. Seventeen cases were classified wrong leading to accuracy ~89%.

Translation of questions is similar to NTCIR-5. To provide redundancy, three translation routes are used to create three Chinese forms from each English question: (a) use Systran MT (http://www.systransoft.com/index.html) to translate a raw English question \( Q^E \) to un-segmented Chinese \( q^{C_1} \); (b) expand each English question with the most frequent fifteen terms from top-ranked snippets returned via Google searching with \( Q^E \). Then translate the expanded question by Systran with segmented Chinese output. Leftover terms are further translated by our web-based translation, which is oriented to entities \( q^{C_2} \). All three forms are concatenated into one final query \( q^C \) for retrieval. The procedure is designed to give more weight to original question translation as well as its entities.

Once a Chinese query \( q^C \) is created for an English question, indexing, retrieval and candidate extraction are performed as in monolingual. The parameters and some of the functions for the evidence factors for ranking the candidates are slightly different, and are discussed below.

3.1 Ranking of Answer Candidates

Because of the translation step, cross-lingual QA is much more uncertain compared to the monolingual case. To reflect this, some of the parameters used in the five evidence factors for ranking candidate answers are modified accordingly based on experimentation with the NTCIR-5 tasks for training.

(a) Categorical Evidence: \( V_c \)

As in Sec.2.4(a), the meaning of level assignment for \( V_c \) is similar but the following values \((70, 30, 5, 5, 1)\) seem to be more helpful for the cross-lingual situation. In general \( V_c \) is assigned a lower value even though there is agreement between \( C_Q \) and \( \text{tag}(a) \).

(b) In-Question/In-Web Evidence: \( V_w \)

As discussed earlier, an English question was expanded with 15 terms from the web and translated to form part of a query. These 15 terms not only enrich the question context, but they might contain an answer (in English) to the question as well. In previous monolingual work, investigators do indirect QA by identifying an answer from these web pages (or other external sources), and later finding a document or sentence that contains this answer. Here, we do not attempt to extract answers from the web pages. However, every candidate (extracted from the retrieved sentences) is compared with the expansion terms and separately with the original translated query \( q^{C_1} \cup q^{C_2} \), and assigned a value \( V_w \) as follows:

\[
V_w = \begin{cases} 
1; & \text{if (candidate occurs in translated original question)} \\
0; & \text{else if (candidate occurs in translated expansion terms)} \\
2; & \text{otherwise}
\end{cases}
\]

We assume that if a candidate occurs in the set of expanded terms, it has a higher chance of being an
answer, but that an answer should not appear in an original (translated) question statement.

(c) Proximity Evidence: V_p
Corresponding to Sec.2.4(c), the form V_p of the proximity evidence formula is the same, but the f() and g() functions are changed: f() = (match-length)^1.5, g() = (distance-from-candidate)^2, and a_p = 0.25.

(d) Similarity Evidence: V_s
The formula for V_s also remains the same, but h() = rank^0.5, and a_s = 2.0.

(e) Frequency Evidence V_f
Here, the influence of the frequency factor is lessened by changing the parameter to a_f = 0.1.

In NTCIR-5, it was observed that frequency evidence V_f was not useful for CLQA. During NTCIR-6, we have only used the four factors V_c*V_w*V_p*V_s for ranking the answer candidates.

3.2 English-Chinese CLQA Results

Results of our E-C CLQA experiments appear in Table 3. Official runs are pircs-E-C-04 to -06 where only the Top1 answers are considered (Rows: Top1, Right[1], Right&U[1]). They all employed the parameter values as discussed in Sec.3.1, differing only in the retrieval depth and whether ‘mpn’ is used or not. (Systran translation also leaves ASCII numbers unchanged in its output). The best run is E-C-06 (with d=80 and no ‘mpn’) having 38 out of 150 questions correct, i.e. an accuracy and MRR of .2533. This is better than using d=100 (E-C-04). Unlike monolingual case where ‘mpn’ helps, if ‘mpn’ were used with d=80 (not shown), the accuracy drops to .24. If all five evidence factors were used (not shown), Top1 correct remains 38, but Top2-5 improves to (16, 6, 5, 3) from (10, 9, 5, 3: E-C-u-12) showing that unlike last year, the frequency factor can lead to better quality at the higher ranks.

Comparing with our best monolingual run (C-C-02), this CLQA run returns only 38/63 ~60% of monolingual – a substantial drop in effectiveness due to translation. If one also counts unsupported answers as correct (Right&U[1]), E-C-06 picks up an additional 4 (total 42) correct, leading to an accuracy+U/MRR+U of .28, and a ratio to monolingual of ~63%.

Table 3 also shows results when Top2-5 answers are also considered for evaluation. E-C-06 run accumulates an additional 27 (total 65) correct and supported answers for a Top5 effectiveness of .4333 and MRR .3190. If unsupported but good answers are also counted as correct (Table 3, bottom half), Top5+U and MRR+U achieve .48 and .3546 respectively - one finds the correct answers at less than the 3rd position on average. For CLQA, more correct answers are returned compared with monolingual as one relaxes to Top5, and the ‘%mono’ values are ~ 74-77%. Unlike C-C, Top1-3 captures only about 2/3 of the extra good answers.

Fig. 3 shows the variation of effectiveness vs. retrieval depth d for runs with the pircs-E-C-06/u-12
Table 4: E-C CLQA Results with Sentence Support - Category Breakdown

<table>
<thead>
<tr>
<th>Query Type</th>
<th>#Q</th>
<th>1% good</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>1-5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Artifact</td>
<td>7</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>Date</td>
<td>39</td>
<td>16</td>
<td>41</td>
<td>2</td>
<td>3</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>Location</td>
<td>16</td>
<td>7</td>
<td>44</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>9</td>
</tr>
<tr>
<td>Money</td>
<td>8</td>
<td>3</td>
<td>38</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>Numex</td>
<td>11</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Organization</td>
<td>16</td>
<td>5</td>
<td>31</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>Percent</td>
<td>4</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Person</td>
<td>47</td>
<td>5</td>
<td>11</td>
<td>4</td>
<td>4</td>
<td>5</td>
<td>0</td>
</tr>
<tr>
<td>Time</td>
<td>2</td>
<td>2</td>
<td>100</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>2</td>
</tr>
</tbody>
</table>

parameters and no ‘mpn’. The bimodal behavior is less noticeable. It appears that d ~ 80-90 is good for runs with supported answers (solid lines), although for runs that include unsupported answers (dotted lines) d ~ 20-40 is better.

Table 4 displays the accuracy of the different answer categories. Five categories (organization, location, date, time, money) show reasonable performance of >30%. While ‘artifact, numex and percent’ each returns zero, ‘person’ achieves only 11%. It is not clear why ‘person’ behaves so poorly.

4 Post-Evaluation Experiments

After evaluation results were known, we made some new runs to study the effects of including different evidence factors for answer candidate ranking, and the effects of question classification. We have used parameters that give our best effectiveness for NTCIR-6 but adding ‘mpn’ for monolingual (i.e. d=25, mpn), and using five evidence factors during extraction and ‘mpn’ for cross-lingual (i.e. d=80, mpn). ‘mpn’ improves results a little with 5 evidence factors, in contrast with 4. These are not submitted.

4.1 Effects of Evidence Factors

The frequency evidence factor was found to be not useful for CLQA in NTCIR-5. For NTCIR-6 however, if all five evidence factors are included, the E-C-u-12 (d=80) run would return (38, 16, 7, 3, 3) correct supported answers for Top1-5 (not submitted). This gives slightly better MRR (.3312) compared to .3212 shown in Table 3. Fig.4 (monolingual) and Fig.5 (CLQA) show some of the synergistic effects when the evidence factors are combined. When each factor is used by itself, the accuracy is low – Fig.4 shows that for monolingual case, the best single evidence factor is proximity V_p (24 correct supported answers) and category V_c (21). Using similarity factor V_s only gives 8 correct. As we add other factors one by one, it progresses to 64. The largest jumps occur from 8 to 24 when in-question evidence V_w is added to similarity factor (sw), and from 24 to 49 when category evidence V_c is further added (swc). Frequency factor returns 0 by itself, but improves results from 57 to 64 when it is used in combination with ‘swc’. For E-C CLQA Fig.5, proximity (8) and similarity (5) evidence have the best results when used singly. However, category evidence improves ‘sw’ results from 9 to 23, and proximity further improves ‘swc’ results to 36.

4.2 Perfect Question Classification

Since our question classification accuracy is about 86% for Chinese and ~89% for English, we investigate how much better results may be obtained if we have a perfect classification algorithm. The result is shown in Table 5 and represented as the last bar in Figs.4 and 5. For monolingual case, perfect
Perfect-classification Runs

<table>
<thead>
<tr>
<th>Top1-5: Right with Sentence Support</th>
<th>Mono (25,mpn)</th>
<th>EC-CLQA (80,mpn)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Right[1]</td>
<td>72</td>
<td>39</td>
</tr>
<tr>
<td>Right[2]</td>
<td>7</td>
<td>15</td>
</tr>
<tr>
<td>Right[3]</td>
<td>10</td>
<td>8</td>
</tr>
<tr>
<td>Total Top1-3</td>
<td>89</td>
<td>62</td>
</tr>
<tr>
<td>Right[4]</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>Right[5]</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>Total Top1-5</td>
<td>93</td>
<td>67</td>
</tr>
<tr>
<td>Top1 Accuracy</td>
<td>.48</td>
<td>2.600</td>
</tr>
<tr>
<td>MRR</td>
<td>.5319</td>
<td>.3354</td>
</tr>
<tr>
<td>Top5</td>
<td>.62</td>
<td>.4467</td>
</tr>
</tbody>
</table>

Top1-5: Right including Unsupported

| Right&U[1]                        | 77            | 45               |
| Right&U[2]                        | 8             | 17               |
| Right&U[3]                        | 10            | 9                |
| Total Top1-3                      | 95            | 71               |
| Right&U[4]                        | 4             | 3                |
| Right&U[5]                        | 1             | 4                |
| Total Top1-5                      | 99            | 78               |
| Top1+U Accur.                     | .5133         | .3000            |
| MRR+U                             | .5702         | .3870            |
| Top5+U                            | .6667         | .5200            |

Table 5: Mono & CLQA Results - Perfect Question Classification

classification brings an additional 8 correct supported answers to Top1 (total 72), and an accuracy of .48. For CLQA, it is a surprise that better classification does not help Top1 at all, but improves Top2-5 results. The Top1-5 number of correct supported answers are (39, 15, 7, 3, 2) compared with (39, 14, 6, 3, 2) with 89% non-perfect classification accuracy.

5 Conclusion

We performed monolingual Chinese factoid QA experiments using only entity extraction software with our PIRCS retrieval and ad hoc answer ranking formulae and parameters. Results with 150 questions showed that it is possible to attain an accuracy of 0.42 (63/150) for Top1 answers with sentence support. A similar strategy together with MT and our entity-oriented web-based translation is applied to English-Chinese CLQA. The corresponding accuracy is .2533 (38/150), which is only 60% of monolingual result due to translation loss. If Top5 answers are considered, the number of questions with correct answers improves to 88 (from 63) and 66 respectively. E-C CLQA improves much more then, achieving 75% (66/88) of monolingual result.

Of the five evidence factors, frequency is the least consistent. All five evidence factors contribute to our best monolingual results. For CLQA, the frequency factor does not contribute to Top1 results, but can improve Top2-5 values.

Our question classification accuracy is 86% for Chinese and 89% for English. Perfect classification does not help Top1 results for E-C CLQA, but improves monolingual Top1 accuracy from .42 to .48.

References