Using Semantic Relations with World Knowledge for Question Answering

Ka Kan Lo and **Wai Lam**
Department of Systems Engineering and Engineering Management,
The Chinese University of Hong Kong,
Hong Kong
{kklo,wlam}@se.cuhk.edu.hk

Abstract

Two research directions are to be explored in realizing our group's TREC QA system in 2006. The first one is to investigate the possibilities of applying linguistically sophisticated grammatical framework in tackling the real-world natural language processing task such as question answering. The other is to exploit the possible world's entities and relations as described in online encyclopedia in adding redundancy and hidden relations as those contained in the TREC corpus where the entities and relations are only implicitly mentioned and related. Our focus is on the factoid and list question as these two types of questions benefit greatly from our proposed method. We do include an experimental component in handling the ”other” question type.

1 Introduction and Previous Work

Current approaches in QA task have applied natural language processing or computational linguistic techniques in various ways. Earlier works, while not using the precise language methodologies, shed some lights on the inherent complexity of the language task. Simple regular expressions have been a common practice in matching the particular question patterns to an answer type. Though some successes have been achieved in the earlier phase of QA, as the complexity of the tasks increases and the relationships between the question and answer share far less similarities in the surface order of the words, the performance of this method quickly declines in tackling more real-world type questions.

Later approach in applying natural language processing method requires more sophisticated grammar approach in capturing the syntactic relations between the entities in a sentence. One example is the use of dependency grammar as in Minipar. This type of grammatical framework is chosen because of the rather simplicity of the backbone describing the different linguistic phenomena and efficient mechanism in parsing. The relatively high efficiency in parsing is critical to the real-world natural language task as several thousands of documents have to be processed in a short time interval. In addition, instead of pure part-of-speech information as obtained by probabilistic CFG parser, the Minipar produces a more detailed analysis about the dependency relations that benefits greatly in the answer extractions.

In our approach, we use a more linguistically sophisticated grammar to analyze the candidate sentences to look for potential answers. The potential to use more detailed analysis of natural language sentences is explored. Based on this analysis, the sentences are broken into more detailed syntactic and semantic description for facilitating the answer extraction process.

Besides the adoption of more complex grammatical framework, the possibilities of using external resources are also explored. Current approaches in QA have also made use of the external resources to improve the redundancy of the potential answers. Some systems make use of a search engine to expand the queries to the TREC corpus and to extract a relevant set of documents from web for answer extraction. Other approach involved using WordNet or FrameNet to provide ontological relations between entities within the corpus. WordNet and FrameNet are chosen because of the very detail analysis of the senses of different words and the relationships between different senses. More details relations such as hypernyms, hyponyms, and synonyms to compile a network of relations of words for question processing and answer extractions. Some other approaches also used the ontology dictionary and online encyclopedia to improve the redundancy of the answers.

In our approach, we choose the online encyclopedia as a major source of external resources. Besides improving redundancy of the answers by providing by a larger set of relevant documents, the inter-relationships between the relevant documents are also extracted for the analysis of questions and answer extraction.

The rest of the paper is organized as follows. In Section 2, we discuss the architecture of the QA system. In Section 3 we detail the method of using grammatical framework to analyze the candidate sentences and answer extractions. Section 4 describes the usage of the external resources in providing a more inter-relationships among entities. Section 5 describes the definitional question processing. Section 6 gives the conclusion.

2 System Description

The general architecture of the system is given in figure 1

The various components of the QA system are described in this section.

2.1 Indexing

The major document resources used in the QA system are the TREC corpus and the Wikipedia corpus. The indexes for these two resources are built up using the LEMUR toolkit. The Wikipedia corpus, which is obtained in May, is first transformed into the TREC corpus format and the extra tags within the corpus are removed to facilitate the parsing of the documents by the LEMUR parsers. A total of 2GB and 3.5GB indexes were built for these two corpora respectively.

2.2 Question Preprocessing and Query Expansion

The question preprocessing phase involves question rewriting and question classification. These two operations use rule-based approach to substitute missing or inadequate information into the questions and classify the questions into one of the sixty-two different types of questions for further processing. For the question preprocessing, some common tasks to be performed are: 1PE. Pronoun matching, 2. Term Expansion, 3. Event-type question special processing and so on. These operations increase the recall of the retrieved document sets in the later and the precision of the candidate sentence extractions. The question is also classified based on whether the question is asking for a person, corporate, time, geographical location such as countries or cities,... , natural places such as hills, seas, lakes,... , time whether year, month, day, date, basic items or description such as profession, colors,... , and special type such as reasons, causes,...

The query expansion phase expands the questions to a more detailed query. In expanding the query, we use the search engine API. The preprocessed question is tokenized and fed to the search engine. The top 10 results are retrieved. The texts in the top results are extracted and form a candidate set of terms.

Besides using the web resources, the preprocessed question is also fed to the index of Wikipedia resource. The texts from the retrieved top 10 documents are extracted. Only terms with external linkages are extracted and merged with the previous set of terms.

2.3 Document Retrieval

The combined set of terms is ranked on the co-occurrence value of terms. Only the terms in the top 50% of the co-occurrence value is used in document retrieval.

The expanded query is fed into the document retrieval based on the LEMUR index. In retrieving the documents, we select the top 100 documents for each question in each series. For the Wikipedia data set, the top 10 documents extracted are passed on to the next phase.

2.4 Candidate Answer Sentences Processing

Quite a number of tasks are performed in this phase before the real matching process. The retrieved documents from both indexes are segmented into sentences. Named entity recognition is performed on both document sets to discover the special entities. For the Wikipedia data set, we also make use of the links to determine the potential candidates for entities.

The extracted set of sentences, typically from two thousands to sixteen thousands, is then ranked based on the word density ranking method.

The final set of fifty sentences are extracted with their respective previous one and next sentences extracted. Thus, a total of fifty core sentences and one hundred peripheral sentences are extracted for answer extraction.

2.5 Wikipedia data processing

The extracted 10 documents from Wikipedia are processed based on Section 4. The result is a network of inter-relationship between different entities that may be the final answer in a candidate question. Sometimes, the 10 documents are not closely inter-related with each others. Extracted documents are extracted and processed to fill up the missing links.

2.6 Answer extraction

The fifty core extracted sentences and the one hundred peripheral sentences from the Candidate Answer Sentences processing are passed to the extraction module. In this module, the question and candidate answers are parsed, the semantic relations are obtained. The semantic relations between the questions and the answer sentences are then compared based on the level of consistency as well as the linkages from the Wikipedia.

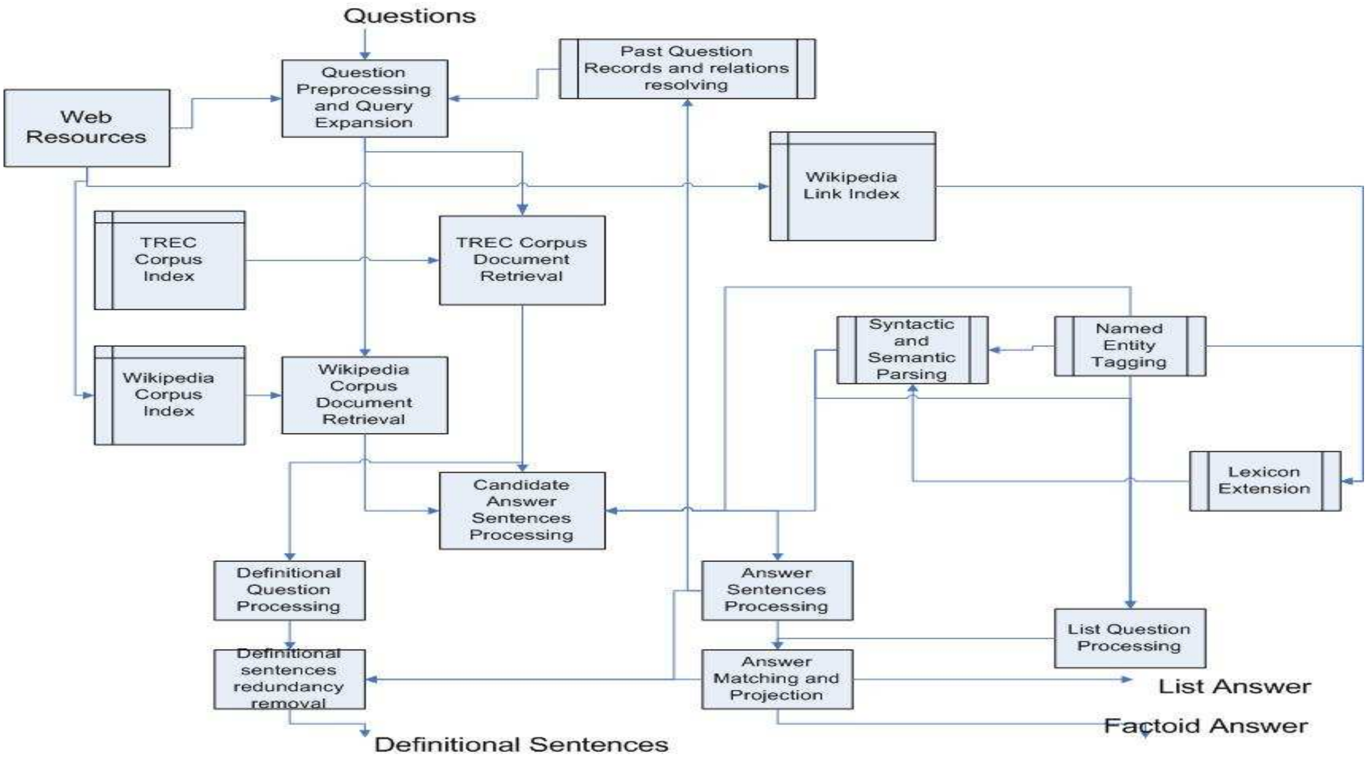

Questions
Question Preprocessing and Query Expansion
Past Question Records and relations resolving
Web Resources
Wikipedia Link Index
TREC Corpus Index
TREC Corpus Document Retrieval
Syntactic and Semantic Parsing
Named Entity Tagging
Wikipedia Corpus Index
Wikipedia Corpus Document Retrieval
Lexicon Extension
Candidate Answer Sentences Processing
Definitional Question Processing
Answer Sentences Processing
List Question Processing
Definitional sentences redundancy removal
Answer Matching and Projection
List Answer
Factoid Answer
Definitional Sentences

Figure 1: General Architecture of the QA system

2.7 List Question Processing

The list question processing module uses the same answer extraction as the normal factoid questions, i.e. using different phases such as question preprocessing, query expansion, candidate answer sentences processing and answer extraction, with an additional mechanism to select the highest ranked list of answers.

2.8 Definition Question Processing

We also use an experimental definition question processing mechanism in handling the definitional questions. Due to the incompatibility of the current design of our QA system to this type of question, we are trying to get some experiences in dealing with the definitional question in this year's QA task.

3 Extracting answers for factoid question and list questions

The goal of this task is to extract the best answers from the corpus to answer the question. The actual operations of our QA system require this task to be integrated with the tasks of the Wikipedia data processing. For ease of demonstration, we concentrate on the use of grammatical framework in this section. The PET parser (0; 0), which uses the English Resource Grammar (0) as the source of lexicons, does not have a wide coverage in covering most of the lexical items in the texts. We thus have to modify the original PET parser a little bit and populate the lexicon system with a larger set of lexicons.

3.1 Parsing

To populate the lexicon, the first problem we need to solve is the multiword expression. The first approach to deduce the basic syntactic type of a particular chunk of words is the use of a named entity tagger to discover a chunk of words as well as whether a particular chunk of words belong to the class of person, location, organization, time and currency. These chunk of words are then stored as a single unit for later processing. The other approach is to check the particular words expression against the Wikipedia to discover whether a particular chunk of words are belonging to some real world entities. This reduces the burden of the further grammatical processing.

Consider the question q141.5:

Who is Warren Moon's agent?
Instead, Moon flew to Los Angeles, where he huddled with his agent, Leigh Steinberg, who has apparently convinced Moon that the Seahawks owe him a ton of career-ending cash.

The named entity tagger discovers that the answer ”Leigh Steinberg” is of the type ”person” and the Wikipedia can locate this entry as a chunk of words. Using the named entity tagger sometimes fails to find the entire name which may be very important in the final answer selection process. For example in question q153.3:

What was Hitchcock’s first movie?
#NYT19990808.0091#Then there’s the TV anthology series, “Alfred Hitchcock Presents,” which ran between 1955 and 1965 (“ Gooood evening, laaadies and gen-tell-men”). #
Prev: #1#NYT19990808.0091#Books? During the past two decades, Hitchcock has become the filmmaker most written about, a publishing phenomenon second only to Princes Di and the Kennedys biographies, memoirs, critical studies, trivia and quiz books, chronicles of the making of “Psycho” and “Vertigo.”#
Next: #0#NYT19990808.0091#It has been regularly in syndication ever since, engendering its own books, fan following and Web sites. #

The phrase ”Alfred Hitchcock Presents”, which is an answer to the question, cannot be detected by the name entity tagger. However, the chunk is detected with the external resources. Besides the entities expansion, there are other words that both resources cannot cover. In this case, WordNet is consulted to find the relevant sense to be used in the parsing process. The expanded lexicon system is then used for parsing. The resulting parse tree and the semantic representation is given below figure 2 and figure 3 for the example sentence ”What position did Moon play in professional football?” from q141.1.

The most important data structure that the questions and the answers are compared is the semantic representation.

3.2 Semantic Ranking

From the semantic representation, each word consists of a list of argument roles, namely ARG0, ARG1 and ARG2. Though, we performed some experiments on the possibilities of using other parameters to further improve the accuracy of the extraction process and with improvements in the results, however, the overall experimental perfor-

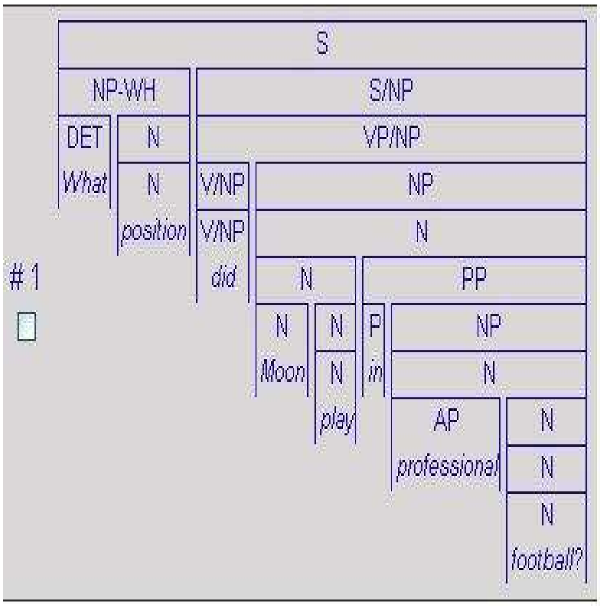
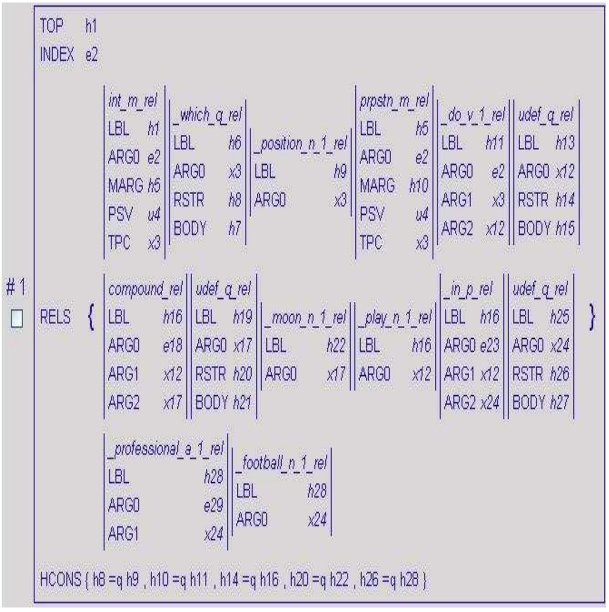


Figure 2: Syntactic parse result



```
     TOP    h1
     INDEX  e2

            int_m_rel    _which_q_rel                  prpstn_m_rel  _do_v_1_rel  udef_q_rel
            LBL    h1    LBL     h6   _position_n_1_rel LBL     h5   LBL    h11   LBL   h13
            ARG0   e2    ARG0    x3   LBL        h9    ARG0     e2   ARG0   e2    ARG0  x12
            MARG   h5    RSTR    h8   ARG0       x3    MARG    h10   ARG1   x3    RSTR  h14
            PSV    u4    BODY    h7                    PSV      u4   ARG2   x12   BODY  h15
            TPC    x3                                  TPC      x3

# 1         compound_rel udef_q_rel                                _in_p_rel    udef_q_rel
□    RELS { LBL    h16   LBL   h19  _moon_n_1_rel _play_n_1_rel    LBL   h16    LBL   h25  }
            ARG0   e18   ARG0  x17  LBL     h22   LBL     h16      ARG0  e23    ARG0  x24
            ARG1   x12   RSTR  h20  ARG0    x17   ARG0    x12      ARG1  x12    RSTR  h26
            ARG2   x17   BODY  h21                                 ARG2  x24    BODY  h27

            _professional_a_1_rel
            LBL           h28   _football_n_1_rel
            ARG0          e29   LBL        h28
            ARG1          x24   ARG0       x24

     HCONS { h8 =q h9 , h10 =q h11 , h14 =q h16 , h20 =q h22 , h26 =q h28 }
```

Figure 3: Semantic relations extracted

mance is not as high as expected and considerable computational overhead is added to the extraction process. We thus neglect these parameters and focus on the argument role to do the extraction.

Based on the similarity between the argument roles, the core and peripheral sentences are ranked for answer projection and extraction.

Similar strategies are also used to rank the sentences from the Wikipedia data set.

3.3 Answer Projection and extractions

A set of answer nuggets are obtained from the major corpus and the Wikipedia data set. The answer set is extracted based on whether the nuggets match the entity type expected by the questions, the consistency between the nuggets obtained from the major corpus and the Wikipedia data set and the consistency of the argument roles filled by a particular name entity.

4 Uses of external resources for better answer extraction

In addition to the grammatical process, we also try to process the external resources systematically to help us project the answer back to the corpus. The only external resources that we processed in this year's QA task are the Wikipedia. The link structure between the Wikipedia actually provides a projection of the world event. Though the total world projected by the Wikipedia sometimes cannot match precisely to the world projected by the text in the TREC corpus, the systematic use of this type of resource can provide a better estimation and some hidden facts in the TREC corpus that may be critical in obtaining a more accurate answer.

Consider the question series q210.1

What government position did she assume in 1993?
#NYT20000617.0162#Did agents with the Bureau of Alcohol, Tobacco and Firearms fire indiscriminately at the Davidians on Feb. 28, 1993 when five group members died?#
Prev:#0#13551.703125#NYT20000617.0162# The civil trial will examine four areas of potential government liability regarding 81 of the Davidians who died at Mount Carmel. #
Next:#2#1.259611#NYT20000617.0162# Did the FBI demolish Mount Carmel prematurely and not in accordance with U.S. Attorney General Janet Reno's directive? #

Though the 3 sentences fragment do say something about Janet Reno in 1993, however, the relations between Janet Reno, the year 1993 and the position she had were unclear.

However, following our link algorithm on this entry text, the relations between ”Janet Reno”, ”1993”, ”Attorney General” are related in this way:

”Attorney General”
”United States Attorney General
”Janet Reno March 12, 1993 January 20, 2001 Bill Clinton”

Thus, the question and candidate answers and the world can be linked up for more accurate answer extraction.

The process of systematic extraction is as follows: From the question and candidate answer sentences, a set of entities are extracted by the earlier result of name entity tagger and Wikipedia lookup. The links are then traced based on the link index we build up for the Wikipedia text. The potential answer sets are those entries with forward and backward links connected, i.e. a cycle is formed.

5 Definitional Question Processing

Since the method we used for the factoid and list questions seem to be not applicable to the definitional type question, we just use the simple terms matching process and redundancy removal to extract the definitional sentences. A set of 500 documents are retrieved based on the question series to the TREC document index. The sentences within the 500 documents are then processed to discover a set of commonly co-occurring set of words. A vector is formed based on the set of words with the highest co-occurrence. This approach, using a set of commonly co-occurring words or centroid words is described in (0). The set of sentences within the 500 documents are ranked based on the centroid word using simple vector matching approach. These sets of sentences are then compared against the list of sentences that are extracted for factoid and list answer extraction based on the vector-based matching. The redundancy sentences, which have a high similarity measures with the answer extraction sentences are removed. The top-N sentences, after the removal process, will be extracted as the definitional sentences.

6 Conclusion

We have described our approach in our new QA system. It includes using more sophisticated grammatical framework, the projection and mapping of the real world onto the TREC corpus and a currently rather simple approach in the definitional question processing system. The current approach, which is still quite immature, has a lot of rooms for improvements. The current procedure in extrapolating the lexicon is not adequate and misses a lot of analytical details for further parsing process. Better approach in extrapolating the lexicon is needed. Current approach in using the world resources and projection back to the TREC corpus is insufficient and there is still a large error in tracing the link structure. More robust link relation extraction procedure is expected for better projection onto the TREC corpus. The definitional question processing, which is very primitive in the current approach, has to be upgraded substantially for future QA system.

References

Jinxi Xu Ana. Trec2003 qa at bbn: Answering definitional questions.

Ulrich Callmeier. Pet. a platform for experimentation with efficient hpsg processing techniques. *Journal of Natural Language Engineering*, 6(1):99–108, 2000.

Ann Copestake and Dan Flickinger. An open source grammar development environment and broad-coverage English grammar using HPSG. In *Proceedings of the 2nd International Conference on Language Resources and Evaluation*, Athens, Greece, 2000.

Berthold Crysmann, Anette Frank, Bernd Kiefer, Stefan Muller, Gunter Neumann, Jakub Piskorski, Ulrich Schafer, Melanie Siegel, Hans Uszkoreit, Feiyu Xu, Markus Becker, and Hans-Ulrich Krieger. An integrated architecture for shallow and deep processing.