# Using a Trie-based Structure for Question Analysis

**Luiz Augusto Sangoi Pizzato**
Centre for Language Technology
Macquarie University
2109 Sydney, Australia
pizzato@ics.mq.edu.au
http://www.clt.mq.edu.au

## Abstract

This paper presents an approach for question analysis that defines the question subject and its required answer type by building a trie-based structure from a set of question patterns. The question analysis consists of comparing the question tokens with the path of nodes in the trie. A look-ahead process solve the mismatches of unknown words by assigning a entity-type or semantically linking them with other question words. The developed approach is evaluated using different datasets showing that its performance is comparable with state-of-the-art systems.

## 1 Introduction

When a question is presented to a person, or even to an automatic system, the first task, in order to provide an answer, is to understand the question. The question analysis process may not be very clear for people when answering questions, however for an automatic question answering (QA) system it plays a crucial role.

Acquiring the information embedded in a question is the primary task that allows the system to execute the right commands in order to provide the correct answer to it. According to Moldovan et al. (2003), when the question analysis fails, it is hard or almost impossible for a QA system to perform its task. The importance of the question analysis is very clear in the system of Moldovan et al. (2003) since this task is performed by 5 of the 10 modules that compose their system.

The most common approach for analysing questions is to divide the task into two parts: Finding the question expected answer type, and finding the question focus.

Many systems (Mollá-Aliod, 2003; Chen et al., 2001; Hovy et al., 2000) use a set of handcrafted rules for finding the expected answer type (EAT). Normally the rules are written as regular expressions (RE), while the task of finding the EAT consists of matching questions and REs. Every RE will have an associated EAT that will be assigned to a question if it matches its pattern.

For the task of finding the question focus, the simplest approach is to discard every stopword on the question and to consider the remaining terms as the focus representation.

In the approach described in this paper, the EAT and the question focus are defined using a trie-based structure built from a manually annotated corpus of questions. The structure stores the answer type in every trie node and uses the question words or entity types to link the nodes.

The question analysis method was evaluated over an annotated set of question of an academic domain, over the annotated TREC-2003 questions and over the 6,000 questions of the training/testing set of question of Li and Roth (2002) showing promising results.

This paper addresses a technique used to analyse natural language (NL) questions and its evaluation. Section 2 describes the technique, while Section 3 presents its evaluation. In Section 4 some related work is described. Finally, in Section 5 we present the concluding remarks and some further work.

## 2 Question Analysis

The developed technique for finding the EAT and the focus of the questions is based on a training set of questions. The questions in the training corpus are marked with their EAT and with their entities and entity types.

A training question is delimited by the tag `Q`. The `Q` tag must contain the attribute `AT` telling the EAT of a question. The question may contain entities, and these entities can be marked to help the learning process. For the purposes of presentation, the entity annotation is done in a way similar to the named entity task of past Message Understanding conferences (Grishman

Proceedings of the Australasian Language Technology Workshop 2004
Macquarie University, Sydney, December 8th, 2004. 

Accepted after full paper review

and Sundheim, 1996) by using the ENAMEX tag and its type attribute.

(1) <Q AT='NAME'> Who is the <ENAMEX type="POS">dean</ENAMEX> of <ENAMEX type="ORG">Macquarie University</ENAMEX>?</Q>

Observe that Example 1 informs that 'dean' is a POS (Position) and 'Macquarie University' is an ORG (Organization).

Every question in the training file provides one question pattern. For instance, Example 1 informs that a question matching the RE in Example 2 asks for a name.

(2) Who is the (.+) of (.+)?

Notice that the RE of Example 2 has two groups of variable terms. If a question matches the RE, it is possible to assume that the words inside the groups match the same entity category as the one defined in the question RE. According to Example 1, the Example 2 categories are POS and ORG.

In our technique we use the words matching the non-fixed part of the RE as the question focus, while we define the EAT using the answer type of the RE.

### 2.1 Trie-based Structure

A trie $T(S)$, according to Clément et al. (1998), is a data structure defined by a recursive rule $T(S) = \langle T(S/a_1), T(S/a_2), \ldots, T(S/a_r)\rangle$, where $S$ is an set of strings of the alphabet $A = \{a_j\}_{j=1}^{r}$, and $S/a_n$ is all string of $S$ that starts with $a_n$ stripped their initial letter.

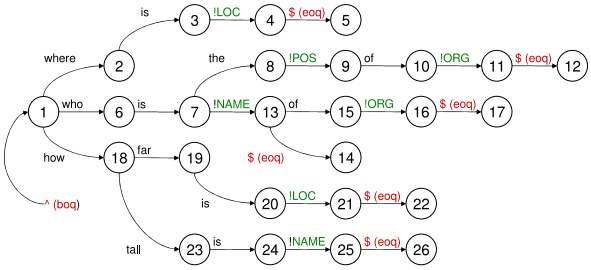
In our question analysis we used a trie-based structure where our 'strings' are the question patterns and our 'alphabet' is the set of question words and entity types.

A question pattern is a representation of the RE where the beginning and the end of question is marked and its non-fixed parts are represented by the entity type. For instance Example 1, would be transformed to:

(3) ^Who is the !POS of !ORG $

The construction of our question trie is similar to the construction of a dictionary trie. However the information stored, the tokens used, and the structure utilisation are different.

In the trie construction phase, every time a node is visited or created, the information regarding the frequency of its EAT is recorded. Since a node in the trie can be reached from different patterns, it is likely that we have a set




| Nodes | Information (EAT,Frequency) |
|---|---|
| 1 | (LOC,1),(NAME,1),(DESC,2),(NUMBER,2) |
| 2-5 | (LOC,1) |
| 6-7 | (NAME,1),(DESC,2) |
| 8-12 | (NAME,1) |
| 13 | (DESC,2) |
| 14-17 | (DESC,1) |
| 18 | (NUMBER,2) |
| 19-26 | (NUMBER,1) |

Figure 1: Trie for the question patterns of Table 1

of frequencies and categories recorded on every node.

Figure 1 shows how the information is structured and recorded in our question trie in case of training the patterns of Table 1. It can be observed that every node in the trie records one or more EAT.

Table 1: Training question/patterns of Figure 1

| Question | Pattern | EAT |
|---|---|---|
| Where is Chile? | ^Where is !LOC$ | LOC |
| Who is the dean of ICS? | ^Who is the !POS of !ORG$ | NAME |
| Who is J. Smith? | ^Who is !NAME$ | DESC |
| Who is J. Smith of ICS? | ^Who is !NAME of !ORG$ | DESC |
| How far is Athens? | ^How far is !LOC$ | NO |
| How tall is Sting? | ^How tall is !NAME$ | NO |

### 2.2 Trie-based Analysis

There are many differences as well as similarities between the utilisation of our trie structure for question analysis and the extraction of indexes from word tries. The first step in the question analysis is to transform the question into the pattern-like format of Example 3. The pattern-like format requires the beginning-of-question and end-of-question marks and if known (by the use of a Gazetteer file) the substitution of some of the question phrases by their entity type.

Using the question' patterns we try to match the first token of the question with the nodes of the trie. If a match is found, then the next token is searched on the nodes linked with the first one. This process continues until there is no more tokens to be examined or the current

Accepted after full paper review

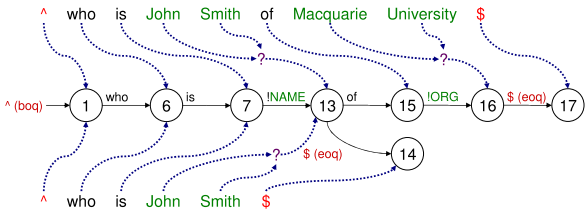



Figure 2: Look-ahead process in the analysis of questions

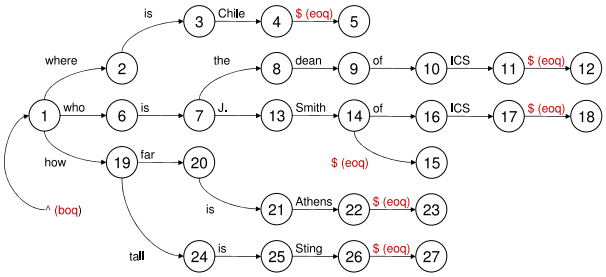



Figure 3: Trie-based structure built without entity information

token can not be matched against the following trie nodes.

This process returns the EAT with the highest frequency of the last visited node. This information will be used as the EAT of the question that was been analysed.

If the current token does not match any following nodes, then a look ahead becomes necessary. In this case the next token is examined over the next nodes of the following nodes. Figure 2 exemplifies the look-ahead process on the analysis of the questions 'Who is John Smith?' and 'Who is John Smith of Macquarie University?' over the trie of Figure 1

The analysis of question 'Who is John Smith?' is done by matching the beginning-of-sentence token and the words 'who' and 'is'. Notice that the words 'John' and 'Smith' and the phrase 'John Smith' were not replaced by their entity type since their condition as names is unknown by the Gazetteer. The word 'John' is not found in the nodes following 'is' (node 13), so the next question word ('Smith') is then searched in those nodes (14 and 15) which are 2 nodes away from the last matched one (node 7). The process continues to search for words in the question in a 2 nodes distance from the last word/node found.

If a match is found, all the words that were not found in previous interaction, are assumed to be of the same type as the node in between the matches. If more than one match is found, the path with the highest frequency will prevail. In this process, the node between the matching words/nodes will define the entity-type of the non-matching phrase on the question pattern.

In the examples of Figure 2, both questions complete the analysis and are assigned a description (DESC) as their EAT. If the process consumes all the tokens of the question and still does not find a match in the nodes, then the last visited node will define the question EAT.

The focus is defined by the entity part of the pattern-like representation of a question. The replacement of some of the question phrases by their entity types can be done before (using the Gazetteer file) or during the utilisation of the trie in the look-ahead process. In both occasions the phrases and their entity types define the question focus. For the questions of Figure 2 the focus would be the 'NAME' 'John Smith' and the 'ORG' 'Macquarie University'.

Our method also considers incomplete matches of question in the trie. If such cases occur, the EAT with the highest frequency of the last visited node will be assigned to the question. For instance, the most frequent EAT of node 6 will be assigned to the question 'Who?' since it is too short to completely traverse the trie. In a similar situation, the question 'Who killed JFK?' cannot be fully matched in the trie and the information of node 6 will define its EAT. Observe that in both cases the last analysed node defines the EAT.

As previous stated, our method requires a training corpus of questions annotated with their EAT and, if possible, with their entities and entity types. The method for finding the EAT does not require the markup of entities. In this case the trie is built only with the information from the words of the questions. Figure 3 shows the question trie constructed from the questions of Table 1 discarding the entity information.

When the entities and entity types are not marked, the analysis of question will still perform the same look-ahead process as demonstrated before. However, in this case, the look-ahead process does not define an entity category but describes an unknown relation between a word in the training questions and another word

or phrase in the question that is been analysed.

To illustrate this situation, consider the question 'Who is the administrative assistant of Macquarie University?'. Since neither 'administrative' nor 'assistant' can be found in the tier of Figure 3, the look-ahead process matches the word 'of' with node 10, assuming that there is a relation between 'administrative assistant' with 'dean'. The same situation will occur with 'Macquarie University' and 'ICS'.

In the current development of our technique, the information about the semantic relations of these words are simply discarded. Further studies are needed to understand where this semantic relations can be used in our QA method.

When the recognition of the entities and their entity types is not possible, the focus is defined by the remaining words in a stopword removing procedure. In some cases this approach finds the same focus words as our entity recognition, however it lacks the information of their entity type.

## 3 Evaluation of the Question Analyser

Our question analysis technique was intrinsically evaluated using a semi-automatically constructed training set of questions. We did not perform any extrinsic evaluation in the sense of Jones and Galliers (1996). That is to say, we did not perform any evaluation of the question analyser over the results in an embedded application such as the question answering task.

The training set contains 1385 randomly selected questions from a set of approximately 40,000 NL questions. The questions were extracted from the JustAsk search engine logs between February 2000 and April 2004. JustAsk is an information retrieval interface to the Macquarie University web site that encourages its users to present queries as full NL questions.

The questions posed in JustAsk are clearly domain dependent, since the search engine is limited to the university domain. Further studies are needed to evaluate how feasible this training set is in questions of different domains.

For the evaluation, we wanted to determine the impact of the size of the training set. For this, we randomly created a training set of $x$ questions and we used the remaining questions for evaluation. To iron out potential idiosyncrasies of the training test we repeated the evaluation $n$ times (normally $n = 200$ but for practical reasons sometimes we used different values)

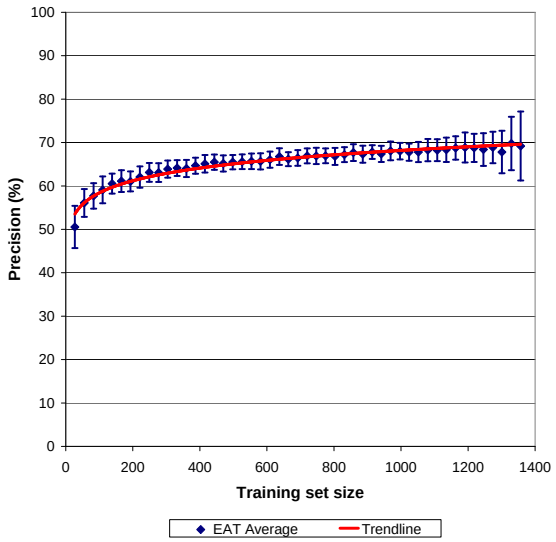



Figure 4: Average results for the EAT

and computed the average of the results, which are shown in Figures 4 and 5

Figure 4 shows a graphical representation of the evaluation of the question analysis over a set of 1385 annotated JustAsk NL questions. It also shows that the EAT precision improves according to the size of the training set. As the size of the training and the verification sets are directly related, it is possible to observe higher standard deviation in the results when few questions are used either for training or for verifying.

We observed that in Figure 4 the precision seems to have a limit in between 70 and 75 percent. In order to measure the hypothetic limit of these measures, we executed a test using the same set of questions for training and for validating the technique. The test showed that the maximum performance when the system was trained and validated with the full set of questions was around 85%.

The test also showed that the maximum performance for finding the EAT degrades when more training questions are provided. This happens because when new questions patterns are introduced, some of them may be similar and present ambiguous information to the overall system. In many cases questions with similar structures require different types of answers. Observe Examples 4 and 5:

(4)
```
<Q AT='NAME'> Who is the
<ENAMEX type='POS'>chair</ENAMEX>
of
<ENAMEX type='EVENT'>ALTW</ENAMEX>
</Q>
```

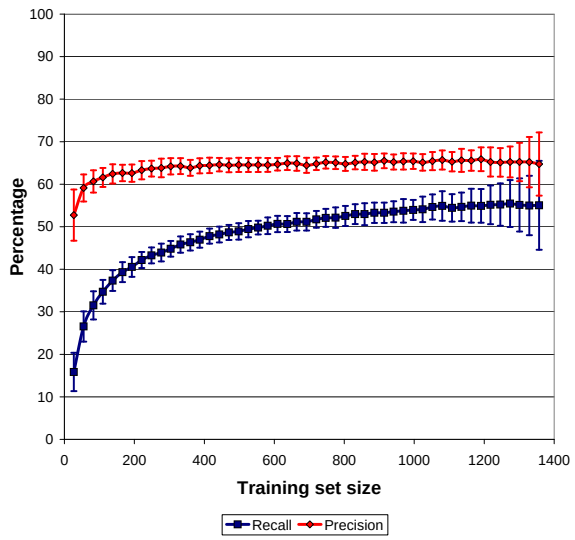



Figure 5: Average results for the question focus

(5)
```
<Q AT='ORG'> Who is the
<ENAMEX type='POS'>sponsor</ENAMEX>
of
<ENAMEX type='EVENT'>ACL</ENAMEX>
</Q>
```

Both examples follow the same pattern (`^Who is the !POS of !EVENT$`), however Example 4 asks for a name of a person while Example 5 requires a name of an organization.

Figure 5 shows the evaluation of the question focus using precision and recall measures. Recall represents the percentage of entities in the verification set that were identified as focus by the question analysis, while the precision measure represents the percentage of entities found that actually existed in the original question.

The evaluation of Figure 5 shows that the performance of the focus identification improves for every new data inserted in the training set. The average of the recall measure increases from less than 20% to more than 50% with less than 600 questions. The results also show that after a few training questions the precision of the discovered entities is kept around 60 and 70 percent for all the training section.

The precision score in Figure 5 gives the impression to have a 65% limit while recall appears to have a limit in the region of 55% and 60%. An estimation of the maximum performance for the entities recognition revealed that the precision value could be as high as 80%, while recall value reaches 85% when all questions used for training are used for validating the technique.

The technique used to assign EATs to questions does not require the markup of entities in the training questions. And because of that, we were able to evaluate the technique on the set of TREC 2003 questions that were manually marked with their EAT information.

The results of this evaluation demonstrated that the precision increases as the size of the training set increases, reaching the mark of 70% with less than 150 training questions and approaching 80% on 400 questions.

To understand if the higher precision of the system in TREC 2003 question was achieved due to the lack of entity information, we tested the EAT precision of the system using the JustAsk training questions with and without the annotation of entities. The idea was to comprehend if the presence of the entities improve or worsen the quality of the EAT analysis. We observed that there were no significant differences between the results, therefore the inclusion or not of entities marks in the training set have to be defined exclusively by the goal of the analysis.

It is clear that the inclusion of entities markup will provide important information about the semantic role of the words in the query focus. However, the cost of marking entities in the question set may not be viable when the question analysis is only used for finding the EAT.

## 4 Related Work

The importance of a good question analysis for QA is clear. The correct EAT identification helps QA process to pinpoint answers by allowing it to focus on a certain answer category. The right question focus provides QA systems with knowledge that helps systems to choose the best sentences to support answers. In this section we discuss some of the techniques used for the task of question analysis.

According to Chen et al. (2001) the EAT recognition falls into two broad groups, those based on lexical categories and those based on answer patterns. The EAT analysis based on lexical categories can be identified by the lexical information present in the questions, while the analysis based on answer patterns are predicted by the recognition of certain question types.

It seems that the most popular approaches for the EAT identification are based on answer patterns. Most works in this group performs the analysis of questions using handcrafted rules

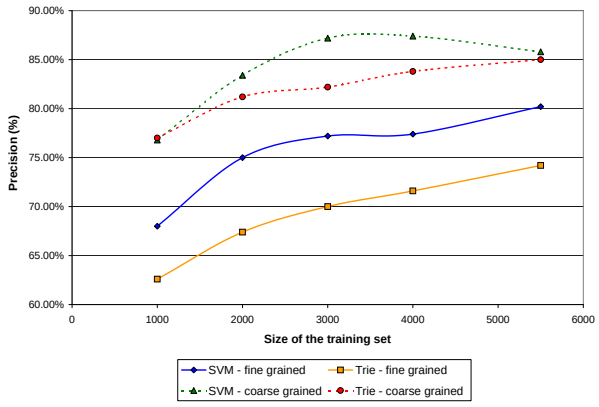



Figure 6: Average results for the trie-based and the SVM approaches

(Mollá-Aliod, 2003; Chen et al., 2001; Hovy et al., 2000).

Hovy et al. (2000) built a QA typology in order to create specific to general EAT. Question patterns were assigned for every answer type, and for those some examples of questions were provided. In a further work Hermjakob (2001) described their intentions of migrating from manual defined rules to automatic ones.

Our system, as described in this paper, uses a rule based approach to automatically build a trie-based question structure. This type of approach has the advantage of being capable of changing domains or even languages by using a different set of training questions.

In order to understand how well our technique performs in comparison to others, we tested our system using the same training/test set of questions used by the LAMP QA system (Zhang and Lee, 2003b).

The LAMP QA system uses a Support Vector Machine (SVM) to classify questions into answer categories. In further work Zhang and Lee (2003a) evaluated their technique using the testing dataset of Li and Roth (2002). Figure 6 compares the results of our trie-based approach with the one using SVM.

The comparison with Zhang and Lee (2003a) technique was made using the same testing dataset and considering the results of Zhang and Lee using bag-of-words features. This comparison shows that SVM provide better results for fine grained answer categories, while for coarse grained answer categories both techniques provide similar results when using the training sets of 1000 questions and 5500 questions.

The comparison shows that our technique provides reasonable result without the need of linguistic resources. And once again we notice that the accuracy of our technique improves when more training data is provided.

With a different approach some systems identify their EAT by using some lexical information of the questions. For instance, the work of Paşca and Harabagiu (2001) uses WordNet (Fellbaum, 1998) to assign a category for its answers. Their system matches questions' keywords with WordNet synsets, and by finding dependencies between synsets, derives an EAT from it.

Paşca and Harabagiu (2001) affirm that their approach for identifying the EAT was successful in 90% of the TREC-9 questions. Their approach for the EAT recognition used the Princeton WordNet along with an answer type taxonomy and a name entity recogniser. Their experiments showed that the use of a large semantic database can help to achieve high quality precision over ambiguous questions stems for finding the questions' EAT.

WordNet has been successfully used in almost every kind of natural language application; undoubtedly it can provide important information to question analysers. For instance, in the QA system of Na et al. (2002) WordNet supports some manually defined questions patterns in the classification of answer categories.

The evaluation of our question analyser shows that we can achieve good results regarding solely in pattern information. We believe that the performance of our system can be boosted by using a hybrid approach, where question patterns are combined with lexical and semantic information.

## 5 Concluding Remarks

This paper presented a method for question analysis that uses a trie-based structure in order to obtain the focus and the expected answer category of a question. The trie-based question analyser was evaluated by using different sets of annotated questions, demonstrating that the developed technique can be used as an alternative to handcrafted RE, since it is a simple method which provides reasonable quality results.

We observed that by increasing the size of the training set our method gets better results. In spite of the fact that the method shows an upper limit in performance, for either recognition of the EAT and the question focus, the results are

not far from the hypothetic maximum value.

It is observed that the hypothetic maximum performance decreases when the training set increases in size. This, as already stated, is due to implicit characteristics of question patterns; however this decrease in quality may be accentuated when poor or no guidelines are presented on the stage of building the training corpus.

Sometimes the job of defining the questions' EAT and their entities is hard even for human annotators. Some questions may have different interpretation on different occasions making the question analysis a challenging task. It is essential that the same decisions are made by the human annotator when dealing with ambiguous questions. Since this problem was only identified during the annotation of JustAsk training questions, our training set may contain some noisy markups. Some further work is needed to determine how this noise degrades the results of the question analysis.

Manual question markup requires not only well defined guidelines but also a great amount of time. The complexity of manually building a training corpus increases when the annotation of named-entities is required. In future work we intend to use the training questions without the markup of the named-entities. We are planning on using the parts of speech (POS) of the questions words and some semantic information from WordNet to assign the question focus and to find out its semantic role.

The extraction of the question focus has not been totally explored yet. For the question analysis on the Macquarie domain, the results for extracting the focus are promising. However, we believe that the combination of POS and semantic information may increase the precision and recall for either focus and the EAT.

To further ensure the effectiveness of the question analyser, we still need to perform an extrinsic analysis in a working question answering environment. Still, the results shown in this paper provide enough evidence that the our question analysis is feasible to be applied in a QA system.

## References

J. Chen, A.R. Diekema, M.D. Taffet, N. McCracken, N. Ercan Ozgencil, O. Yilmazel, and E.D. Liddy. 2001. Question answering: CNLP at the TREC-10 question answering track. In *Proceedings of TREC-2001*, pages 485–494.

J. Clément, P. Flajolet, and B. Vallée. 1998. The analysis of hybrid trie structures. In *Proceedings of the Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 531–539, Philadelphia, PA. SIAM Press.

C. Fellbaum. 1998. *WordNet – An electronic lexical database*. MIT Press, Cambridge, Massachusetts.

R. Grishman and B. Sundheim. 1996. Message understanding conference-6: a brief history. In *Proceedings of the Coling'96*, pages 466–471.

U. Hermjakob. 2001. Parsing and question classification for question answering. In *Proceedings of the Workshop on Open-Domain Question Answering at ACL-2001*, Toulouse, France, July.

E. Hovy, L. Gerber, U. Hermjakob, M. Junk, and C. Y. Lin. 2000. Question answering in webclopedia. In *Proceedings of TREC-9*, pages 655–654.

K. Sparck Jones and J. R. Galliers. 1996. *Evaluating Natural Language Processing Systems: An Analysis and Review*. Springer-Verlag New York, Inc.

X. Li and D. Roth. 2002. Learning question classifiers. In *Proceedings of the COLING-02*, pages 556–562.

D. Moldovan, M. Paşca, S. Harabagiu, and M. Surdeanu. 2003. Performance issues and error analysis in an open-domain question answering system. *ACM Trans. Inf. Syst.*, 21(2):133–154.

D. Mollá-Aliod. 2003. Answerfinder in TREC 2003. In *Proceedings of TREC-2003*.

S.H. Na, I.S. Kang, S.Y. Lee, and J.H. Lee. 2002. Using grammatical relations, answer frequencies and the world wide web for TREC question answering. In *Proceedings of TREC-2002*.

M. Paşca and S. Harabagiu. 2001. High performance question/answering. In *Proceedings of SIGIR'01*, New Orleans, Louisiana, USA. ACM.

D. Zhang and W.S. Lee. 2003a. Question classification using support vector machines. In *Proceedings of the SIGIR-03*, pages 26–32. ACM Press.

D. Zhang and W.S. Lee. 2003b. A web-based question answering system. In *Proceedings of the SMA Annual Symposium 2003*, Singapore.