Question Answering : CNLP at the TREC-10 Question Answering Track

Jiangping Chen, Anne R. Diekema, Mary D. Taffet, Nancy McCracken, Necati Ercan Ozgencil, Ozgur Yilmazel, and Elizabeth D. Liddy

Center for Natural Language Processing
Syracuse University, School of Information Studies
4-206 Center for Science and Technology
Syracuse, NY 1324-4100
www.cnlp.org
{jchen06,diekemar,mdtaffet,njm,neozgenc,oyilmaz,liddy}@syr.edu

Abstract
This paper describes the retrieval experiments for the main task and list task of the TREC-10 question-answering track. The question answering system described automatically finds answers to questions in a large document collection. The system uses a two-stage retrieval approach to answer finding based on matching of named entities, linguistic patterns, and keywords. In answering a question, the system carries out a detailed query analysis that produces a logical query representation, an indication of the question focus, and answer clue words.

1. Introduction

Question-answering systems retrieve answers rather than documents in response to a user's question. In the TREC question-answering track a number of question-answering systems attempt to answer a predefined list of questions by using a previously determined set of documents. The research is carried out in an unrestricted domain.

The CNLP question answering system uses a two-stage retrieval approach to answer-finding based on matching of named entities, linguistic patterns, and keywords. In answering a question, the system carries out a detailed query analysis that produces a logical query representation, an indication of the question focus, and answer clue words. Then the information is passed on to answer finding modules, which take the documents, retrieved in the first stage, for further processing and answer finding. The answer finding module uses three separate strategies to determine the correct answer. Two strategies are based on question focus, and the third strategy, based on keywords, is used when the question focus is not found or when the first two strategies fail to identify potential answers. A detailed system overview can be found in section 3.

2. Problem description

CNLP participated in two of the three QA track tasks: the main task and the list task. The main task is a continuation of last year's QA track in that systems are required to answer 500 short, fact-based questions. Two new aspects were introduced this year: unanswerable questions (with no answer present in the collection), and the notion of an answer confidence level. Systems needed to identify unanswerable questions as such, in order for them to be counted as correct. For the confidence level systems needed to state the rank of their final answer or state that they were unsure about their answer. For each question, up to five ranked answer responses were permitted, with the most likely answer ranked first. The maximum length of the answer string for a submitted run was 50 bytes. A response to a question consisted of the question number, the document ID of the document containing the answer, rank, run name, and the answer string itself.

The list task questions are similar to those of the main task but include an indication as to how many answer instances needed to be provided for an answer to be considered complete. A response to a list task question consisted of an unordered list with each line containing the question number, the document ID of the document containing an answer instance, and the answer string itself. The length of the list or the number of answer instances for each question is specified in the question. As in the main task, the maximum length of each answer string is 50 bytes. The different answer instances could be found within

single documents or across multiple documents or a combination of both. There was no guarantee that all requested answer instances could indeed be found in the collection.

Answers to both main task and list task questions had to be retrieved automatically from approximately 3 gigabytes of data. Sources of the data were: AP newswire 1988-1990 (728 Mb), Wall Street Journal 1987-1992 (509 Mb), San Jose Mercury News 1991 (287 Mb), Financial Times 1991-1994 (564 Mb), Los Angeles Times 1989, 1990 (475 Mb), and Foreign Broadcast Information Service 1996 (470 Mb). The submitted answer strings for all tasks were evaluated by NIST's human assessors for correctness. [10] Examples of questions for both the main task and list task can be found in table 1.

TREC-10 QA questions
Main task questions: How much does the human adult female brain weigh? , Who was the first governor of Alaska? , When was Rosa Parks born? , Where is the Mason/Dixon line? , Why is a ladybug helpful? , Where is Milan? , In which state would you find the Catskill Mountains? , What are invertebrates?
List task questions: Name 2 U.S. dams that have fish ladders. , What are 6 names of navigational satellites? , Who are 6 actors who have played Tevye in "Fiddler on the Roof"? , Name 20 countries that produce coffee.

Table 1. Examples of TREC-10 questions.

3. System overview

The CNLP question-answering system consists of four different processes: question processing, document processing, paragraph finding, and answer finding. Each of the processes is described below.

3.1 Question processing

Question processing has two major parts – conversion of questions into a logical query representation and question focus recognition. Our L2L (Language-to-Logic) module was used this year to convert the query into a logical representation suitable for keyword matching and weighting in our answer finder module (see section 3.3. and section 3.4). Last year we used our L2L module for first-stage retrieval but this year we relied solely on the ranked list of documents retrieved and provided by NIST. L2L was modified this year to also include query expansion for nouns and verbs found in WordNet 1.6 [8]. Based on the parts-of-speech of the question words, the system added all related synonyms of the first, most frequently used sense (see example at the end of this section) to the L2L representation.

Question focus recognition is performed in order to identify the type of answer expected. Expected answers fall into two broad groups – those based on lexical categories, and those based on answer patterns. Expected answers based on lexical categories can be identified from the terms used in the question. For example, in the question "What river flows between Fargo, North Dakota and Moorhead, Minnesota?", we identify that the questioner is looking for the name of a river. The expected answer type, and therefore the question focus, is *river*. Expected answers based on answer patterns are predicted by the recognition of certain question types. If the question is recognized as a definition question, then the answer sentence is likely to include one of several patterns, such as apposition, presence of a form of the verb *be*, etc.

The question focus recognition routine extracted four elements – the question focus, the lexical answer clue, the number of answers required (used for the list task only), and the confidence level (not fully implemented). In an effort to improve question focus recognition this year, we trained the Brill part-of-speech tagger [2] on questions from TREC 8, TREC 9 and HowStuffWorks. [7] The resulting rules were used to tag the TREC 10 questions. The tagged questions were then run through the Collins parser [3] [4] for a full parse.

There are three steps to question focus assignment. In the first step, the question type is determined using predefined search patterns based on regular expressions. There are 7 special question types (acronym, counterpart, definition, famous, standfor, synonym, why) and 7 standard question types (name-a, name-of, where, when, what/which, how). If a special question type is recognized, then the question type becomes the question focus. Second, the parsed question is examined to extract the lexical answer clue (word or

phrase) using the predefined search patterns. In the third step, which applies only to standard question types, the lexical answer clue is used to assign the question focus based on lexical categories where possible. Table 2 is a review of the questions types. Predefined search patterns were developed for these question types:

	Question type	**# of search patterns**	**Example question**
Standard question types	Name-a	3	Name a food high in zinc. (TREC 10, question 1268)
	Name-of	2	What is the name of Neil Armstrong's wife? (TREC 10, question 1007)
	Where	10	Where is John Wayne airport? (TREC 10, question 922)
	When	9	When is the official first day of summer? (TREC 10, question 1331)
	What/Which	14	What is the capital of Mongolia? (TREC 10, question 1050)
	Who	9	Who lived in the Neuschwanstein castle? (TREC 10, question 1281)
	How	12	How tall is the Gateway Arch in St. Louis, MO? (TREC 10, question 971)
Special question types	Acronym	4	What is the abbreviation for Texas? (TREC 10, question 1172)
	Counterpart	2	What is the Islamic counterpart to the Red Cross? (TREC 9, question 454)
	Definition	7	What is autism? (TREC 10, question 903)
	Famous	5	Why is Jane Goodall famous? (TREC 9, question 748)
	Standfor	4	What does the technical term ISDN mean? (TREC 10, question 1219)
	Synonym	3	What is the colorful Korean traditional dress called? (TREC 10, question 1151)
	Why	1	Why does the moon turn orange? (TREC 10, question 902)

Table 2. Question types

Additional processing performed by the question focus assignment routine includes the extraction of the number of answers required (used for the list task only), and assignment of a confidence level. The number of answers required was extracted based on the predefined search patterns for each question type. The confidence level assigned ranged from 0 to 5, with 5 being the highest level of confidence in the question focus. If the question focus could not be determined, the confidence level was 0. Otherwise, the confidence level was set at a value ranging up to 5 depending on the certainty of the question focus. Due to the short time available for development, confidence level assignment was only partially implemented for TREC 10 and therefore not used in the experiments.

The output resulting from the L2L module and the Question Focus recognition module is passed on to the paragraph finding module, the answer candidate recognition module, and the answer formatting module. A standard question type (in this case what/which), will produce the following output for the question “What is the deepest lake in the US?”:

Logical representation:	deep* lake* +US ("United States" "United States of America" America U.S. USA U.S.A.)
Query focus:	lake#deepest lake#2#5
Tagged:	<sentence sid="s0"> whatlWP belVBZ thelDT <CN> deeplJJS lakelNN </CN> inlIN thelDT <NP cat="cntry" id="0"> USlNP </NP> ?l. </sentence>

A special question type (in this case definition), will produce the following output for the question “Who is Duke Ellington?”:

Logical representation:	+Duke* +Ellington*
Query focus:	def#Duke Ellington#2#5
Tagged:	<sentence sid="s0"> whol WP belVBZ <NP cat="per" id="0"> DukelNP EllingtonlNP </NP> ?l. </sentence>

As can be seen in the examples above, expansions from WordNet are enclosed in parentheses, and the four elements in the question focus are separated by '#'.

3.2 Document processing

For document retrieval, we used the ranked document list as provided by NIST. The top 200 documents from the list for each question were extracted from the TREC collection as the source documents for paragraph finding.

3.3 Paragraph finding

In the paragraph finding stage, we aim to select the most relevant paragraphs from the top 200 retrieved documents from the first stage retrieval step. Paragraph selection was based on keyword occurrences in the paragraphs. Although we used the same strategy as last year to identify the paragraphs, we decided to experiment with the selection process itself. For one set of runs we took the original document and divided it up into paragraphs, based on textual clues. After selecting the top 300 most relevant paragraphs we tag only those paragraphs. This approach is identical to our TREC9 approach and these runs are labeled "PAR" (paragraph tagging). For the other set of runs we tagged the original document first, then divided it up into paragraphs from which the top 300 paragraphs were selected. These runs are labeled "DOC" (document tagging). Paragraph detection is no longer based on orthographic clues (i.e. indentations) for the "DOC" runs because this information is removed during the tagging process. The tagged document is divided into several sentence groups based on a pre assigned value that specifies the approximate number of words in each sentence group.

We hypothesized that tagging the whole document versus isolated paragraphs should provide better named entity identification. Named entities are often referred to in their full form early in a document, only to be reduced to a shorter form later on. When an isolated paragraph is presented to our system for tagging, the context information of the preceding paragraphs is not available for entity categorization, thus hindering tagging performance. The complete documents as well as the individual paragraphs were part-of-speech tagged and categorized by <!**meta**Marker>™ using CNLP's categorization rules.[1] The quality of selected paragraphs and the system's categorization capabilities directly impact later processing such as answer finding.

3.4 Answer finding

The answer finding process (see sections below) takes the tagged paragraphs from the paragraph finding stage (for "DOC" as well as "PAR" runs) and identifies different paragraph windows within each paragraph. A weighting scheme was used to identify the most promising paragraph window for each paragraph. These paragraph windows were then used to find answer candidates based on the question focus or additional clue words. All answer candidates were weighted and the top 5 (main task) or top *n* (list task) were selected. The answer finding process expanded answer finding strategies without making major changes to the weighting strategy.

3.4.1 Paragraph-window identification and selection

Paragraph windows were selected by examining each occurrence of a question keyword in a paragraph. Each occurrence of a keyword in relation to the other question keywords was considered to be a paragraph window. A keyword that occurred multiple times thus resulted in multiple paragraph windows, one for each occurrence. A weight for each window was determined by the position of the keywords in the window and the distance between them. An alternative weighting formula was used for single-word questions. The window with the highest score was selected to represent that paragraph. The process was repeated for all 300 paragraphs resulting in an ordered list of paragraph windows - all potentially containing the answer to the question.

3.4.2 Answer candidate identification

This year we focused on expanding the answer candidate identification ability of the system by changing the answer finding strategies and adjusting our weighting schemes based on the TREC9 question set.

Answer candidate identification involves three separate strategies. Two strategies are based on question focus, and the third strategy, based on keywords, is used when the question focus is not found or when the first two strategies fail to identify potential answers. The two question focus strategies include search for a specific lexical category in the case of standard question types and search for a specific answer pattern in the case of special question types (see section 3.1). Which strategy is initially employed for a particular question is based on the value found in the question focus element in the question focus line. If the question focus value matches one of the special question types, then the specific answer pattern strategy is used. If the question focus has a value of "unknown", the third strategy involving keywords is invoked as a fallback. For all other values of the question focus element, the specific lexical category strategy is employed. For a discussion of the specific lexical category strategy and the keyword strategy see our TREC 9 paper. [5] For each special question type (acronym, counterpart, definition, famous, standfor, synonym, why), one or more answer patterns have been identified and defined in the answer candidate identification routine.

3.4.3 Answer-candidate scoring and answer selection

The system used a weighting scheme to assign a weight to each answer candidate. Although we intended to change the weighting scheme to accommodate the new answer finding strategies we ran out of time. The weight was based on the keywords (presence, order, and distance), whether the answer candidate matched the question focus, and punctuation near the answer candidate.
This resulted in a pool of at least 300 candidates for each question. A new unique-answer-identifier module removed duplicate answers from the answer-candidate list. The top 5 highest scoring answer candidates were selected as the final answers for each question for the main task. The required number of answers, identified during question processing, determined the number of answers for the list task questions. The answer strings were formatted according to NIST specifications.

4. Results

We submitted four runs for the TREC10 QA track: two runs for the main task and two runs for the list task. Each run name can be parsed into four components: 1) organization name, 2) trec, 3) tagging approach (see section 3.3), and 4) task.[1]

4.1 Main task results

Averages over 492 questions[2] (strict evaluation):	**SUT10DOCMT**	**SUT10PARMT**
Mean reciprocal rank	0.148	0.218
Questions with no correct answer found	381 (77.4 %)	332 (67.5 %)
Questions with rank above the median	80 (16.3 %)	117 (23.8 %)
Questions with rank on the median	345 (70.1 %)	329 (66.9 %)
Questions with rank below the median.	67 (13.6 %)	46 (9.3 %)
Correctly Answered NIL questions	0 (out of 3)	0 (out of 3)

Table 3. Question answering results for the main task.

The evaluation measure for the main task (see Table 3) is the mean reciprocal answer rank. For each question, a reciprocal answer rank is determined by evaluating the top five ranked answers starting with one. The reciprocal answer rank is the reciprocal of the rank of the first correct answer. If there is no

[1] SU = Syracuse University, T10 = TREC10, DOC = tag entire document / PAR = tag individual paragraphs, MT = main task / LT = list task

[2] The initial question set of 500 questions was reduced to 492 questions after 8 questions were discarded by the National Institute for Standards and Technology.

correct answer among the top five, the reciprocal rank is zero. Since there are only five possible ranks, the mean reciprocal answer ranks can be 1, 0.5, 0.33, 0.25, 0.2, or 0. The mean reciprocal answer ranks for all the questions are summed together and divided by the total number of questions to get the mean reciprocal rank for each system run.

4.2 List task results

Averages over 25 questions	SUT10DOCLT	SUT10PARLT
Average Accuracy	0.25	0.33
Questions with no correct answer found	4 (16 %)	5 (20 %)
Questions above the median	13 (52 %)	15 (60 %)
Questions on the median	9 (36 %	7 (28 %)
Questions below the median	3 (12 %)	3 (12 %)

Table 4. Question answering results for the list task.

The evaluation measure for the list task (see Table 4) is average accuracy. For each question accuracy is determined by the number of distinct correct answers over the target number of instances to retrieve. Accuracy for all the questions is summed together and divided by the total number of questions to get the average accuracy.

5. Analysis

The main task analysis examines: (5.1) retrieval performance of first stage retrieval based on the ranked list provided by NIST, (5.2) the Language-to-Logic module, (5.3) question focus assignment, (5.4) query expansion, and (5.5) the difference between the tagged document and tagged paragraph run performance. The list task analysis (5.6) examines list task performance, instance assignment, and the difference between the tagged document and tagged paragraph run performance.

5.1 First stage retrieval

As mentioned previously, we used the ranked document list as provided by NIST for first stage retrieval. The retrieved lists were created using the PRISE system [6]. For TREC9 NIST used the SMART [9] information retrieval system (see Table 5).

Top 200 results	TREC9	TREC10
Questions without any retrieved documents	0	0
Questions without any relevant retrieved documents	48	32
Questions for which there are no relevance judgments	20	48
Questions with relevant retrieved documents	625	420
Total number of questions	693	500
Total number of documents retrieved	134,600	90,400
Number of known relevant documents	7,963	4,465
Total number of relevant documents retrieved	6,014	2,966
Average precision	0.29	0.23

Table 5. First stage retrieval performance.

Compared to last year's retrieval results, both the number of known relevant documents as well as the average number of retrieved relevant documents for each question decreased. The TREC10 retrieval results might have increased the difficulty of finding correct answers.

5.2 Question representation

A logical representation of the question is created in the question processing stage (see section 3.1). The question representation analysis of this year is based on the main task tagged "PAR" run (SUT10PARMT). We noticed that there were much more short questions this year than the previous two years. Even after query expansion, our system still produced 45 (9 %) single word queries and 64 (12.8 %) two-word queries. Many of these questions are "What/Who is/are/was/were" questions which asked for a definition of a person or a thing. Short queries, although represented correctly, may lead to failure in answer finding

because the current weighting strategy has not been adapted to them. After excluding short queries, 73 (14.6 %) questions had various representation problems. The major query representation problems include keyword selection problems; part-of-speech errors; and misplaced wildcards (see Table 6).

Problem count	Problems with description
30	*Keyword selection problems* Content words such as numbers were erroneously filtered out or truncated, or inappropriate words were selected
16	*Part-of-speech tagging errors* Wrong tags led to incorrect morphological processing and query expansion error
13	*Misplaced wildcards* Wildcards placed in the wrong place of single words created bad stems

Table 6. Question representation problems.

Compared with the query representation of last year, the system has improved the part-of-speech tagging, but did worse on keyword selection. Some important numbers, such as the number in question "What city has the zip code of 35824? " were filtered out by the system, which had a negative impact on answer finding.

Our conclusion of last year held true - query representation problems only accounted for part of the failure of answer finding. The "PAR" run contained 160 questions that did find the correct answer: 53 (33 %) were short queries, and 19 (12 %) had various query representation problems. The procedure we developed for answer candidate identification helped finding answers for short queries. However, the system did not find the correct answers for most of the questions even when the query representations were correct. Further analysis is needed to identify why this is the case.

5.3 Question focus

As described in section 3.1, we determined the question focus based on special question patterns and lexical answer clues. The question focus analysis is based on the main task "PAR" run (SUT10PARMT). Out of 492 answerable questions, our system determined a question focus for 365 (74.17%) of the questions, more than 10 percent better than TREC-9 (see Table 7). [5] Our efforts to improve focus recognition aided in this increase. Out of these 365 questions, 322 questions (88.2 %) had a correct focus, and 43 questions (11.8 %) had an incorrect focus. Not only did we find a question focus for a greater percentage of questions this year, we also found the *correct* focus for a greater percentage of questions as well. For 127 (25.8 %) questions, our system could not determine a focus.

	Correct question focus	Incorrect question focus	No determinable question Focus
Rank 1	68 (21.1 %)	7 (16.3 %)	3 (2.4 %)
Rank 2	27 (8.4 %)	2 (4.7 %)	2 (1.6 %)
Rank 3	15 (4.7 %)	1 (2.3 %)	2 (1.6 %)
Rank 4	12 (3.7 %)	2 (4.7 %)	5 (3.9 %)
Rank 5	12 (3.7 %)	1 (2.3 %)	1 (0.8 %)
Rank 0	188 (58.4 %)	30 (69.8 %)	114 (89.8 %)
Total	322	43	127

Table 7. Answer rank distribution of question focus status.

An analysis of the special question types (see Table 8) shows that some of the special question routines (definition, standfor) aided in finding the answer. Our ability to find the answers for definition type questions in particular is improved over last year. But since the majority of special question types still failed to find a correct answer, more work is needed.

	Acronym	Definition	Standfor	Synonym	Why
Rank 1		19 (19.4 %)	4 (40.0 %)		
Rank 2		7 (7.1 %)			
Rank 3		9 (9.2 %)			
Rank 4		3 (3.1 %)			
Rank 5		5 (5.1 %)			
Rank 0	1 (100 %)	55 (56.1 %)	6 (60.0 %)	5 (100 %)	4 (100.0 %)
Total	1	98	10	5	4

Table 8. Analysis of Special question types[3]

An analysis of lexical answer clues (see Table 9) shows that having the correct lexical answer clue aids in finding the correct question focus.

	Correct question focus	Incorrect question focus	No determinable question focus
Correct Lexical Answer Clue	295 (91.6 %)	70 (64.8 %)	55 (88.7 %)
Incorrect Lexical Answer Clue	27 (8.4 %)	38 (35.2 %)	7 (11.3 %)
Total = 492	322	108	62

Table 9. Lexical Answer Clue vs. Question Focus

In summary, our efforts to improve focus recognition led to a greater percentage of both identified question focus and correctly identified question focus. Having a question focus is clearly important for finding the answer, as 89.8 % (114/127) of the questions with no determinable focus failed to find an answer. Finding the correct lexical answer clues aids in finding the correct question focus. Special question processing helps, but needs improvement.

Since the majority of the questions with a correct focus (188/322 = 58.4 %) did not retrieve an answer, we need to examine this finding in more detail.

5.4 Effects of query expansion

As discussed in section 3.1, we used WordNet 1.6 to expand nouns and verbs in the questions this year. Experiments using the TREC9 questions showed that the expansion helped find more relevant paragraphs, but whether it helped in locating the final answer within those paragraphs was not investigated. Query terms added from WordNet were found in 109 out of 160 (68 %) questions with correct answers in our paragraph run SUT10PARMT.

Query expansion had an additional, positive, impact. It actually provided correct answers for some short queries. For the question "*What does the acronym NASA stand for?* ", the phrase "National Aeronautics and Space Administration", was added to the L2L representation. This feature has been used in our procedure for identifying answer candidates for some question types.

5.5 Document tagging versus paragraph tagging

Contrary to our expectation, the "DOC" run (see section 3.3) did not achieve better performance, but did worse than the "PAR" run (see Table 3). This held true for both the main task and the list task. Following is the comparison of the two runs for main task (see Table 10).

[3] This analysis is based solely on the special question types identified as such; there were a total of 151 special questions.

RunID	# of correct answer	Rank 1	Rank 2	Rank 3	Rank 4	Rank 5
Document tagging (SUT10DOCMT)	111	51	22	20	8	10
Paragraph tagging (SUT10PARMT)	160	78	31	18	19	14

Table 10. Comparison of correct answers found by document run and paragraph run.

We hypothesized that the low performance of the document run might be caused by a lack of system testing due to time constraints. The analysis provided a good opportunity to find system bugs as well as evaluate our approach. We noticed that in most cases where the two runs got the right answers at the same rank, they found the answers in different documents or different paragraphs. A careful examination of the results for the first 103 questions (questions 894 to 996) demonstrated that the following sources contributed to the poor performance of the "DOC" run and the difference between the two runs:

1. A bug in the paragraph finding program truncated some documents when they were split into paragraphs (see section 3.3). The impact of this bug was minor.
2. The keyword weighting strategy used for paragraph finding inadvertently differed slightly between runs, which led to different scores even when the same answer strings were found. The influence of this difference was minor because it did not cause a change in rank.
3. The sentence alignment procedure truncated part of the texts for some documents. The word alignment procedure occasionally failed to record some of the keywords in the paragraph window, which threw out some paragraphs with the correct answers and dramatically changed the weighting score for some answer candidates. The alignment problems were the major cause of the low performance of the document run and the difference between the two runs.
4. The size of the paragraphs also played a role in making the two runs different. In the "PAR" run, we identified paragraphs according to text indentation while the "DOC" run uses a predefined value (400 bytes) to group sentences into paragraphs. Normally the sentence groups are longer than the natural paragraphs. The difference in length changed the position of paragraph windows and led to different scores for the same candidates.

After fixing the bugs and adjusting the alignment procedures (sources 1 and 3), we ran the "DOC" run again and achieved comparable results between the two runs. For the first 103 questions, both runs found correct answers for 30 questions out of which 23 were identical.

We also compared the tagging and categorization between complete documents and individual paragraphs. No difference between the two was found in this analysis. It might be that the TREC10 questions did not bring out the need for context information in tagging. This issue will need further investigation. Ultimately we need to decide between these two approaches.

5.6 List task evaluation

Both list task runs (SUT10PARLT and SUT10DOCLT) are based on the same question processing output. The list task analysis examines the performance of the answer instance identification as well as the reasons for the large performance difference between main and list tasks.

A special feature was added to our question processing module this year to handle the extraction of the number of desired instances from the list questions. Analysis revealed that for 22 (88%) of the questions, the number of instances was determined correctly. For three questions the program could not determine the correct number of instances so it defaulted to 2 instances (enough instances to make up a list). Out of the 22 questions that provided the right number of instances none of the questions managed to get all of the desired answers correct.

The system seemed to perform better on the list task than on the main task (see Tables 3 and 4). For the SUT10PARLT run only 20% of the questions could not be answered versus 67.5% in the main task counterpart run (SUT10PARMT). In observing the questions themselves it appears that the list task

questions are more straightforward compared to the more complicated main task questions where 151 questions required more advanced linguistic pattern searches. The fact that the questions seem to be easier is reflected in the performance of the focus assignment module for the list task. Out of 25 list questions, 13 questions had a correct focus assignment, 3 questions had a wrong focus assignment, and for 9 questions the system correctly indicated that the focus was unknown. For the list task 88 % of the questions had a correct focus assignment versus 78% questions in the main task. Two out of the three questions with the wrong focus assignment were of identical form (*Name n people who/from …*) and both indicated the answer should be a number instead of a person. The error is due to a clue in the focus program dealing with *how many people* questions.

6. Conclusions and future research

The expansion of our question processing module clearly improved the accuracy of our focus assignment although there are still a large number of questions for which the system did not provide the correct answer. It appears that tagging the entire document before splitting it into paragraphs versus splitting it into paragraphs before tagging does not make a lot of difference. The decision on what tagging approach to take will depend on processing speed.

After the TREC10 experiments it is clear that a lot of work remains to be done. Our analysis shows that a one-size-fits-all approach to answer-finding does not work well. The system needs alternative answer-finding strategies for different question types and the means to differentiate between these question types. These different strategies also imply more advanced weighting schemes than are currently implemented. Our work on answer confidence level assignment needs to be completed and refined. The confidence level work will also include the ability to decide whether an answer can indeed by provided. In addition the system also needs to be adapted to deal with the context specific task (the third TREC Q&A track task) where each answer provides contextual information to help answering the next (related) question.

References

[1] <!metaMarker>™ . http:// http://www.solutions-united.com/products_information.html
[2] Brill, Eric. (1995). Transformation-Based Error-Driven Learning and Natural Language Processing: A Case Study in Part of Speech Tagging. Computational Linguistics. Available at: <http://www.cs.jhu.edu/~brill/CompLing95.ps>.
[3] Collins, Michael. (1996). A New Statistical Parser based on Bigram Lexical Dependencies. Proceedings of the 34th Annual Meeting of the ACL, Santa Cruz. Available at <http://www.research.att.com/~mcollins/papers/acl9629.ps>.
[4] Collins, Michael. (1997). Three Generative, Lexicalised Models for Statistical Parsing. Proceedings of the 35th Annual Meeting of the ACL (jointly with the 8th Conference of the EACL), Madrid. Available at <http://www.research.att.com/~mcollins/papers/paper14.short.ps>.
[5] Diekema, Anne; Liu, Xiaoyong; Chen, Jiangping; Wang, Hudong; McCracken, Nancy; Yilmazel, Ozgur; and Liddy, Elizabeth D. (2000). Question Answering: CNLP at the TREC-9 Question Answering Track. Available at: <http://trec.nist.gov/pubs/trec9/papers/cnlptrec9.pdf>.
[6] Dimmick, D., O'Brien, G., Over, P., and Rodgers, W., Guide to z39.50/prise 2.0: Its installation, use & modification, 1998. http://www-nlpir.nist.gov/works/papers/zp2/zp2.html
[7] HowStuffWorks. http://www.howstuffworks.com/.
[8] Miller, G. (1990). WordNet: An On-line Lexical Database. *International Journal of Lexicography*, Vol. 3, no. 4. Special Issue.
[9] Salton, G. Ed. (1971) *The SMART Retrieval System: Experiments in Automatic Document Processing*. Prentice-Hall, Inc. Englewood Cliffs, NJ. 556p.
[10] Voorhees, Ellen M.; Tice, Dawn M. (1999). The TREC-8 Question Answering Track Evaluation. In: E.M. Voorhees and D.K. Harman (Eds.) *The Eighth Text REtrieval Conference (TREC-8).* 1999, November 17-19; National Institute of Standards and Technology (NIST), Gaithersburg, MD.