QACTIS-based Question Answering at TREC-2005

P. Schone, G. Ciany, R. Cutts*, P. McNamee†, J. Mayfield†, Tom Smith†*

U.S. Department of Defense
Ft. George G. Meade, MD 20755-6000

ABSTRACT

The QACTIS system is being developed for the eventual purpose of providing a user the capability of multilingual question-answering from multimedia. QACTIS was tested at TREC-2005 as a means of identifying its successes and limitations in answering questions specifically from English newswire text as it moves in the direction of multilingual, multimedia question answering. In this paper, we provide a complete overview of those parts of QACTIS which focus specifically on text question-answering, and we analyze the system's performance at TREC-2005.

1. INTRODUCTION

QACTIS (pronounced like "cactus"), which stands for "Question-Answering for Cross-Lingual Text, Image, and Speech," is a research protoype system being developed by the U.S. Department of Defense. The goal of the QACTIS effort is to gain greater understanding of question-answering (QA) as a whole while focusing on multilingual and multimedia issues (which areas have largely been outside of the mainstream of QA research). When complete, the prototype should allow users the ability to ask questions in multiple languages and obtain answers which have been derived from multilingual and/or multimedia sources. However, in this as in the last TREC competition (see Schone, *et al*., 2004), we have, like others, focused our efforts on English newswire text as a means of working to establish a credible English-text-focused system before fully venturing into the largely unexplored (and less-than-critical-mass) areas of multilingual, multimedia QA.

Our current approach to question answering can largely be thought of as a graph-search strategy. In short, this approach automatically converts the incoming question into an indexed and attributed entity-relationship graph which has vertices with missing information (representing the actual information need of the user). QACTIS then uses information retrieval to identify a number of full documents which may be on the subject of the user's question. These top documents are then also converted into attributed entity-relationship graph. Lastly, a search is conducted on this graph to find one or more subgraphs with content which satisfies the previously missing information. We refer to this method as a "Knowledge-Graph Induction" strategy. This approach can be applied to any of the factoid or list questions.

However, this knowledge-graph search strategy can be somewhat costly in terms of compute time. Its average question-answering response takes typically from between 30 to 120 seconds on a 3.0 GHz Linux system. Yet for some types of questions, less complexity is required. On such questions (in particular, "other" and "how many") QACTIS employs a "filter cascade" strategy which initially hypothesizes many possible answers and then applies increasingly restrictive filters to hone in on the most likely answers. These filters consist of template-matching, shallow grammatical rule applications, and other filters which can typically be applied in under ten seconds.

The TREC-2005 QACTIS system observed a small overall performance increase over that of our TREC-2004. QACTIS system. In terms of its best overall system, QACTIS's factoid-answering score increased from 20.4% to 25.4% this year, and its list answering increased from F=0.071 to F=0.105. Given that the median scores across all sites in this year's competition decreased from 17.0% to 15.2% and from F=0.094 down to F=0.053, respectively, it is apparent that the gains QACTIS has observed are real. QACTIS's definition ("other") answerer had a decrease in score decreased from F=0.367 last year to F=0.248; yet F=0.248 was the maximum score for this category in 2005.

In the sections that follow, we illustrate the components of our current system as well as indicating the conditions of and scores from each of our three separate Q&A runs as well as our IR results. Furthermore, we give some insights into the experiments that we conducted while try-

* Dragon Development Corporation, Columbia, MD
*Henggeler Computer Consultants, Columbia, MD
†Johns Hopkins Applies Physics Laboratory, Laurel MD

ing to improve the system, as well as indicating any difficulties. Lastly, we outline the future directions that we plan to undertake between this and the next TREC.

2. SYSTEM DESCRIPTION

Figure 1: System Overview

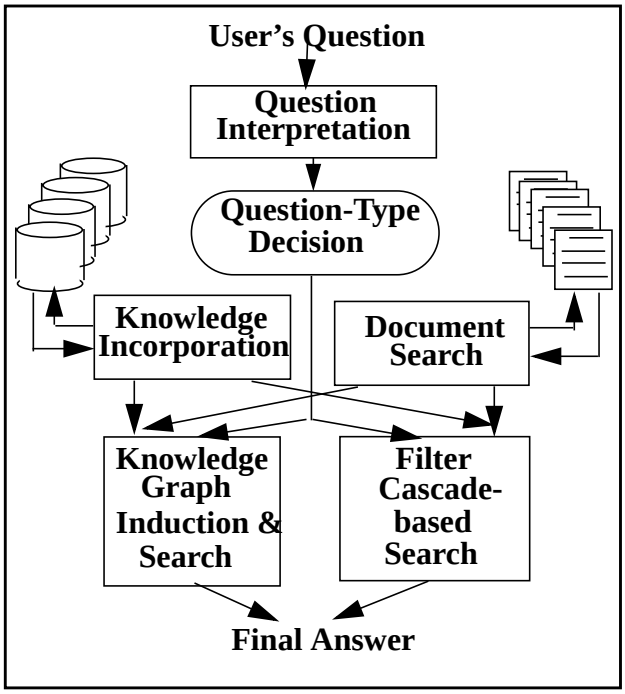


Figure 1 provides a high-level view of QACTIS as it stood in preparation for answering questions of English newswire data for TREC-2005. As was mentioned previously, the QACTIS system was designed with multilimedia and multilingual search and question-answering in mind. The complete system, as it currently stands, comes equipped with such tools as automatic speech-to-text transcription, universal phonetic recognition, and any-language spoken document retrieval [1]. However, we will not describe any of those components in this setting but instead will focus on those pieces which we currently have in place for question-answering in English news text.

Answering questions in QACTIS is a process which can be decomposed into four sub-stages which are comparable to those of other Q&A systems. In particular, these consist of interpreting the question of the user, retrieving documents that may contain answers to the question, identifying knowledge sources that may support answer-finding, and lastly, performing the question-answering itself. Each of these is important for finding appropriate answers, and failures in any one area can result in poor or non-answers. We describe each of these pieces in detail, though, when deemed necessary, we refer the reader to previous documentation for those components that have not changed since TREC-2004 [2].

2.1 Interpreting the User's Question

2.1.1. Handling Question Anaphora

At TREC-2004, a new paradigm was employed by NIST for the question-answering task. Prior to last year's competition, question-answering systems were provided with a large sequence of independent, complete questions. Question-answering systems could answer each question in a stateless fashion, where each question was known to have no relation to any previous questions. In TREC-2004, however, NIST increased the difficulty of the task by treating the questions in small batches which were designed to represent user sessions. In these sessions, it was expected that follow on questions might use anaphora to refer to pieces or answers from preceding questions. Failure to resolve those anaphora would undoubtedly signify a failure to properly answer the question. In TREC-2005, this notion of user sessions continued, so any system would have to first be faced with the task of anaphor resolution before question-answering could begin.

QACTIS made some modest changes to account for user sessions. In last year's proceedings, it was mentioned that QACTIS was designed to tackle five separate kinds of anaphora that might appear in user questions. In particular, (a) these were questions without anaphora; (b) those where a simple anaphor was substituted for the topic; (c) ones where the topic might need to be decomposed (such as by gender) in order to resolve the anaphora; (d) others where neither the topic nor anaphora were implied but not indicated; and (e) questions that referred to the answers to previous questions. It appeared that most of these approaches were reasonably successful last year, so we made few changes. However, when the system thought anaphor resolution would require topic decomposition, there were times in TREC-2004 where it omitted the key components needed for answering the question. Likewise, when previous answers were required (as in case 'e'), we realized that the system was failing to insert previous answers and was breaking. Moreover, we recognized that even if the proper answer had been substituted, there was still only about a 25% that the appropriate answer was inserted.

In some tests this year, we discovered that in many cases, even with no anaphor resolution, the system could do a reasonable job answering the question if the session topic was appended to each question. Therefore, this year, we ensured that the topic in its full form was available in every question, whether as a result of anaphor resolution or as a result of topic-appending. However, in TREC-2004, the topic of the user session was an entity of sorts whereas this year's competition was also focused on events. Topics describing events tended to be longer than the topics for entities, so these were almost never found in anaphor resolution. We have not yet determined how per-

formance would have changed if anaphor resolution would have taken these changes into account, but it is unlikely QACTIS could have been significantly enhanced by work in that area.

2.1.2. Desired Response Type

Since users may ask "factoid," "list," or "other" style questions, the next issue was to resolve how to treat each of these kinds of questions. QACTIS makes a number of policies for each kind of question. These policies reflect the number and length of desired responses, and whether an answer should be automatically trimmed or removed altogether. To be more concrete, we discuss these policy decisions for each of the three desired response types.

Factoid Policy: for factoid question-answering, only a single response can be made. However, there are still decisions as to whether the answer that is returned is sufficiently exact or whether the score for the top returned answer is high enough to be regarded as a valid answer. For every answer, QACTIS returns a score which is related to the odds of the provided answer's being correct. We determined this year, while developing our system on last year's material, that if the score was sufficiently low (in particular, lower that 1.0E-8), QACTIS was better off returning a NIL response than the given response. Additionally, through error analysis of last year's results, we realized that QACTIS was penalized for responses such as "Coach Harold Solomon" instead of "Harold Solomon" because "coach" was in the question (Q27.2). Accordingly, we developed software which could filter candidate answers to allow only those without redundancy or descriptive prepositional phrases to go forward. Rather than "officially" instituting all of these as policy, our three separate runs in TREC-2005 reflected different perspectives on these issues. Our first run implemented both exactness filtering and NIL identification; our second run implemented only NIL identification; and our third run used no filtering.

List Policy: In reality, QACTIS produces a list of potential answers for every question and selects the best answers from that list as factoid answers. In order to tackle list-style questions, it merely needs to be told how many answers to respond with. In the previous year's competition, QACTIS produced a small list of 5-7 answers because the system was less accurate. As the system has improved, and particularly for list-type information, we found that a higher threshold yielded better results. This year, we set the threshold at 15 for every list type question.

Other Policy: Our "other" system had been fairly successful last year by reporting the top 50 most exacting sentences as answers. However, there were two issues that were not addressed last year that became policy this year. Since NIL is never a desired answer for "other" type questions, a null response can only result in a diminished score. Yet we found that in our system from last year, almost 10% of all "other" questions processed through QACTIS had yielded no response. Analysis showed that the system had applied its normal mode of answer-finding and, when no answer was discovered, the system applied no back-off strategy. Also, last year, there was no post-filtering to remove redundancy (eg., duplicate sentences due to the appearance of documents with duplicate content). This year, then, some level of redundancy-checking was used (which will be described later) and a back-off strategy was applied in the absence of a first-pass answer.

2.1.3. What is the Actual Information Need?

In addition to handling response-type policy issues and resolving anaphora, question interpretation also requires identifying the actual information need. Some of the principle system improvements to QACTIS represented the search for and handling of various kinds of content need.

At TREC-2004, QACTIS was equipped primarily with the ability to find answers for questions related to named entities, locations, temporal information, quantities, quantities with units, and some hypo/hypernymic relationships. This year, there were some significant expansions in each of these areas, cross-linking between areas, and addition of question-handling for some new areas. Of particular note, though, are the enhancements that have been made to prepare for answering hypo/hypernym-type questions.

In recent years, TREC has begun to ask many more questions of forms akin to "What rock group...," "What famous scientist..," "What is the range...", "What historical landmark..." Each of these kinds of questions suggest that the information need is a member of some class of words. In our TREC-2004 system, QACTIS would have not known exactly what was being sought and would have looked generally for a noun phrase as the solution to the question. This year, though, we required that the system restrict its answer to a narrow category if it could and thus, only return a rock group, a historical landmark, etc. as an appropriate answer. Our strategy for doing this will be described later. It is expected, though, that this kind of processing resulted in the largest measure of the performance increases that we observed this year.

2.1.4. How to Process a Question Type

The last issue regarding question interpretation is what to do with the question once it has been interpreted. As was indicated in Figure 1, QACTIS has two paths to answering questions depending on the type of question complexity. When the question is determined to be that is quantitative, in particular, is of the form "how many," it will apply a filter cascade for answering finding. Like-

wise, if the question is a definition "other" type, the filter cascade will also be used. QACTIS's knowledge graph induction approach is used for all remaining questions.

2.2 Retrieving Documents

Like many other question-answering systems, QACTIS begins its question-answering phase by first attempting to find documents on the same subject We had made the determination last year that the Lemur system [3] provided comparable performance to what we had developed internally but it was multilingual, faster, and more robust. We therefore continued to use Lemur and process the top 30 documents returned from each question. There have been several recent releases of Lemur, but we did not see any major reasons for upgrading, so our competition system made use of Lemur 2.2.

We conducted a number of information retrieval experiments in hopes of increasing accuracies, but by the competition deadline, these approaches did not result in immediate improvements. One of these strategies was to eliminate duplicate documents. The other was to conduct queries specific to question type that are enhanced with terms that tend to occur in answer-bearing documents. The latter of these processes yielded richer information retrieval results for the IR system on which is was tested, but those results did not outperform Lemur. It is expected that when ported to Lemur, this novel approach will yield improvements. For now, it was not included in QACTIS but was submitted separately by the company that developed the approach as a retrieval-only submission (see [4]).

Redundancy elimination was expected to be beneficial in that it would allow our system to process more unique documents and thus give it a better chance of finding the true answer. We evaluated Lemur's results and we assured that documents with exactly the same score were, in all cases, the same content; and documents with comparable scores often contain overlapping material. Much to our dismay, however, eliminating redundancy reduced the performance of our factoid-answering system on a development set by about 2% absolute. We have considered a number of theories for this phenomenon such as: (1) if a question is proposed by TREC assessors and they want to verify that the answer thereof exists in the corpus, they are more likely to find it if it resides in duplicate documents; and (2) documents with interesting content are more likely to be reappear on various days of a given newspaper or appear in multiple media sources. We have not drawn any particular conclusions, however.

Redundancy removal did seem more appropriate in the "other" question-answering task, though. Given that assessors are looking to find only novel pieces of information, one might assume that redundancy could do nothing but reduce overall performance. The flip side of this is that assessors have to process many files, so they may fail to find nuggests in one passage and yet find the nugget in an exact restatement of that passage. Nonetheless, we did employ redundancy elimination in our "other" processing.

2.3. Identifying Knowledge Sources

As was stated earlier, many of the questions that have been proposed at recent TRECs have been taxonomic in nature, such as "What rock band..." In TREC-2004, QACTIS made extensive use of WordNet [5] as a means of establishing hypo/hypernymic relationships between words. A particular difficulty with this approach is that many of the relationships that are desired are between some word class and some group of named entities. For example, in the version of WordNet that we are using (2.0), the only rock band known by the system is "Beatles." If we are less restrictive and look for the WordNet hypernyms of "band," we only find words with meanings akin to sets, concert and dance bands, striations, stripes, frequency ranges, collars, rings, and ligatures. None of these categories would help answer the question unless one were so lucky as to have a question about The Beatles.

Semantic Forests (see [6]) was a topic identification algorithm developed some years ago which was used in earlier TREC years as a means of information retrieval. This algorithm came equipped with some large electronic dictionaries which had many thousands of additional words added that focused primarily on proper nouns such as "Green Bay Packers" and "Miami Dolphins," which are examples of U.S. football teams. Moreover, the dictionary had been expanded to include over 300 word categories including these proper nouns. We determined that exploiting these classes was very helpful in being able to determine the appropriate answer. However, the classes were incomplete and insufficient in number.

We embarked on a process of trying to distill word classes from Wikipedia [7] into word classes that could supplement the Semantic Forest dictionaries. One would like to be able to mine Wikipedia classes at run time for question answering, but various running conditions may preclude connectivity to Wikipedia or to the Web at large. Since Wikipedia can be copied freely, then, it is perfectly reasonable to capture static snapshots of it and distill categories into an easily readable format. With this being the case, we grew the number of available classes in the system to over 1100 from the previous 300 or so. Likewise, we were able to establish some complete classes of location information by also mining publically available databases [8]. Since these placenames and classes were incrementally added to Semantic Forests' dictionaries, we cannot state the exact performance increase that was observed on development data using this approach, but we

expect that 5-10% absolute is probably not an unreasonable approximation. However, it should be mentioned that as we developed the system in this past year, we tuned the system to the development sets; so only perhaps 1-2% gains were actually realized at competition time.

There is a danger in using such auxiliary information in trying to answer questions. The danger is that QACTIS might find the correct answer in a document because the document discusses the right subject and contains answer words and Semantic Forests knows that those words are the appropriate category. However, assessors may fail to see how the answer words satisfy from that given document satisfy the question and may mark the correct answer as unsupported. For the future, we have begun experimenting with hypernym induction techniques which may help us discover these associations directly from the AQUAINT data (using software described in [10]), but for the present, we were willing to risk "unsupportedness" in favor of being able to find correct answers.

2.4. Answering the Questions

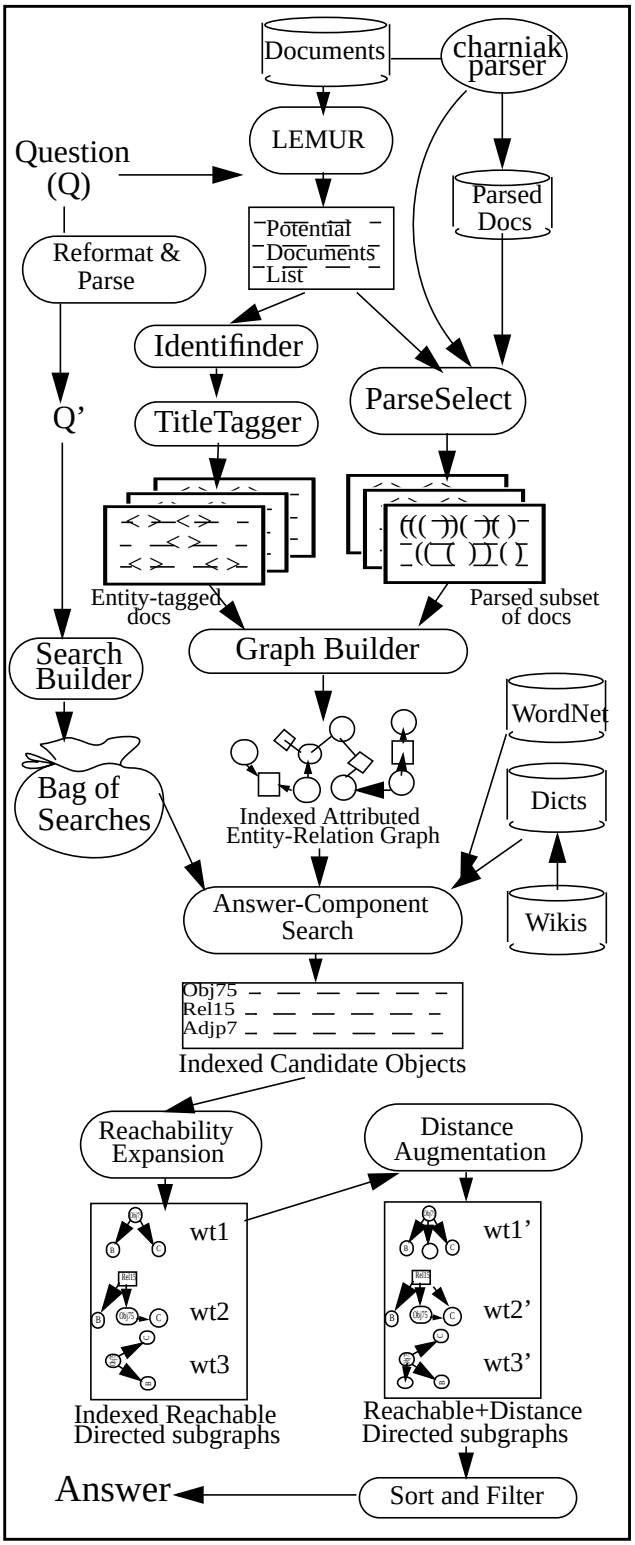
After the introductory steps described above, the final and most crucial step to determine the answer. As was stated previously, QACTIS uses a knowledge-graph induction strategy to answer questions of all forms but "How many" and "other" which are processed by a separate filter cascade strategy. The structures of these systems is largely the same as they were in TREC-2004, so we will only describe the additions and improvements that have been made this year. Yet for the sake of completeness, we do provide a rough overview of the system components and we provide updated illustrations of these systems.

2.4.1. Knowledge-Graph Induction/Search

Figure 2 provides a detailed graphical view of the current methodology employed by the knowledge-graph induction strategy. The basic premise of this approach, as was stated previously and at TREC-2004, is to convert the top *N* potentially relevant documents returned by Lemur into a single, indexed, directed, attributed entity-relationship graph which can be mined to find connected subgraphs containing the desired components of the question.

Synopsis of TREC-2004's Graph Search: In overview of last year's status, this system begins by parsing the TREC collection offline using the Charniak Parser [11] When the user issues a question, top *N* documents are identified by Lemur, their parsed forms are retrieved, stripped of parsing annotations, and run through BBN's Identifinder™ [12] to obtain named entity tags. The question itself is parsed through the Charniak parser, and the information therefrom its turned into a set of "search objects" which the system must try to find the indexed, attributed

Figure 2. Knowledge-Graph Induction/Search



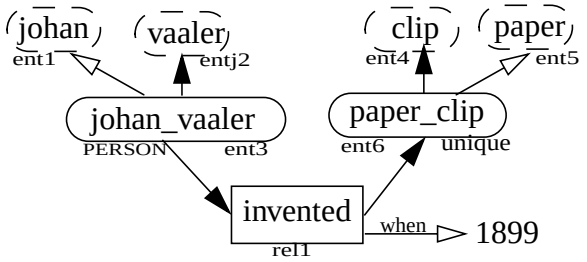
relationship graphs grown from the top *N* documents. To illustrate how this is built, we consider the following example from our earlier documentation. If a news article stated: "Johan Vaaler invented the paper clip 90 years

ago." Charniak's parser would convert in into the annotated string

(S1 (S (NP (NNP Johan) (NNP Vaaler))
(VP (VBD invented)
(NP (DT the) (NN paper) (NN clip))
(ADVP (NP (CD 90) (NNS years)) (RB ago))) (. .)))

Named entity tagging through Identifinder would then indicate that "Johan Vaaler" is a person. The phrase "90 years ago" ia also converted into absolute times (1899) by using the news article's metadata that indicates it was written in 1989. The system next performs some degree of anaphor resolution, and then the graphbuilder builds entities from nouns and noun phrases, relationships from verb phrases, and attributes from the existing quantifiers, prepositional phrases, and adjectives. The end result is akin to that of Figure 3:

Figure 3: Indexed, Attributed Entity-Rel Graph



If the question being posed were "Who invented the paper clip?," the system parses the question, converts "who" into a word suggesting information need: "person.q," and it identifies the major objects which need to be found in the graph: "person.q" and "paper clip" entities and the "invented" relationship. As in TREC-2004, the system then identifies nodes in the attributed entity-relationship graph, and it grows subgraphs therefrom using reachability and distance constraints. These subgraphs are scored for their information content using odds-type weighting, and the subgraph with the highest score is returned as the best answer. The interested reader is referred to the previous QACTIS documentation from last year for additional details of the processes that had been in place at TREC-2004 [2].

Architecture Modifications of Graph Search for 2005: Though our factoid-answering performance is still far below the world's best factoid answering systems, we have confidence that the general approach we are taking is solid and destined for much better future performance. With that in mind, we maintained the same general architecture this year as we had last year but with three useful changes. One of these changes, as was stated in the last section, was the incorporation of taxonomic lists from external sources such as Wikipedia. Another simple architectural change was that the system was designed to automatically parse a document in the event that it tried to mine the top-*N* pre-parsed documents and, for whatever reason, the parse of one or more of those documents was not in the database. The last change was more significant, and this was the development of a "title tagger."

Titles are frequently the appropriate response to a question. For example, "What song...," "What book...", and "What movie...," all represent questions whose response is more than likely a title. However, typical entity tagging attempts to find names of people, places, dates, and organizations ... titles are typically excluded. We therefore developed a rudimentary title tagger that merely looked for strings of words which are delimited before and after with double quotes, which have all non-stopwords being capitalized, and which have at least one capitalized word other than date information. This simple approach helped greatly. For example, question 2345 asks "What are the titles of the books in Sue Grafton's alphabet series of mysteries?" Out of the top 100 answers of the QACTIS system of 2004, not one answer was a title let alone a title of Sue Grafton's books. The QACTIS-2005 system produced the results "O is for Outlaw," "Z is for Zero," "B is for Burglary," "C is for Corpse," "N is for Noose," and "R is for Ruff, Ruff" in 1st, 3rd, 7th, 10th, 22nd, and 24th place, respectively.

Internal Modifications of the Graph Search for 2005: There were a number of other experiments that we conducted and modifications that were made to the system which did not represent architectural changes but did reflect process changes. These could be classified roughly as enhanced anaphor resolution and expanded question-type handling.

Previously, QACTIS's anaphor resolution is a straightforward symbolic method which tries to resolve pronouns and draw associations between definite articles. We tried three strategies for trying to improve this. For one thing, we invested several weeks in developing a system that would do name disambiguation (i.e., Joe Person=Joe=Mr. Person), and we were quite pleased with its ability to resolve these symbols. However, this resulted in a small loss of performance on a development set (two answers were dropped from first place and mean reciprocal rank (MRR) went down by 0.3% absolute). We next tried to reduce the extents to which the system would go to drawing associations between the definite articles. Previously, we allowed the system to resolve anaphors which were up to 30 objects away, but now we reduced that number to 10. This both sped up the system and yielded 0.5% absolute improvement in MRR and two more right answers. Lastly, we allowed for agentive information to satisfy pronouns. More specifically, "The band director

was killed by a car. He ..." would now allow "he" and "band director" to be associated with one another. This form of resolution increased MRR by 0.4% absolute.

The biggest overall system changes, however, were the handling of question types and the related issue of establishing what types a question falls into. Above all, the system was informed that it first needed to turn questions like "What blah ..." or "What was a blah..." into a search for answers that are hyponyms of the "blah" category. The use of the Wikipedia information, as described earlier, helped significantly here. We also added specific handling of the following classes of "what" questions:

- Odds and Percentages
- Causes of death
- Actual locations: is this place really a city as needed?
- Actual times: is this really a day and month as asked?
- Organizations, companies, and businesses
- Titles (as mentioned before)
- Scientific names (incl. consideration of Latin endings)
- Chemical symbols/formulas
- Temperatures
- Ranges
- Wingspans, heights, and other unit-seeking answers
- Salaries, wages, and other money-seeking answers

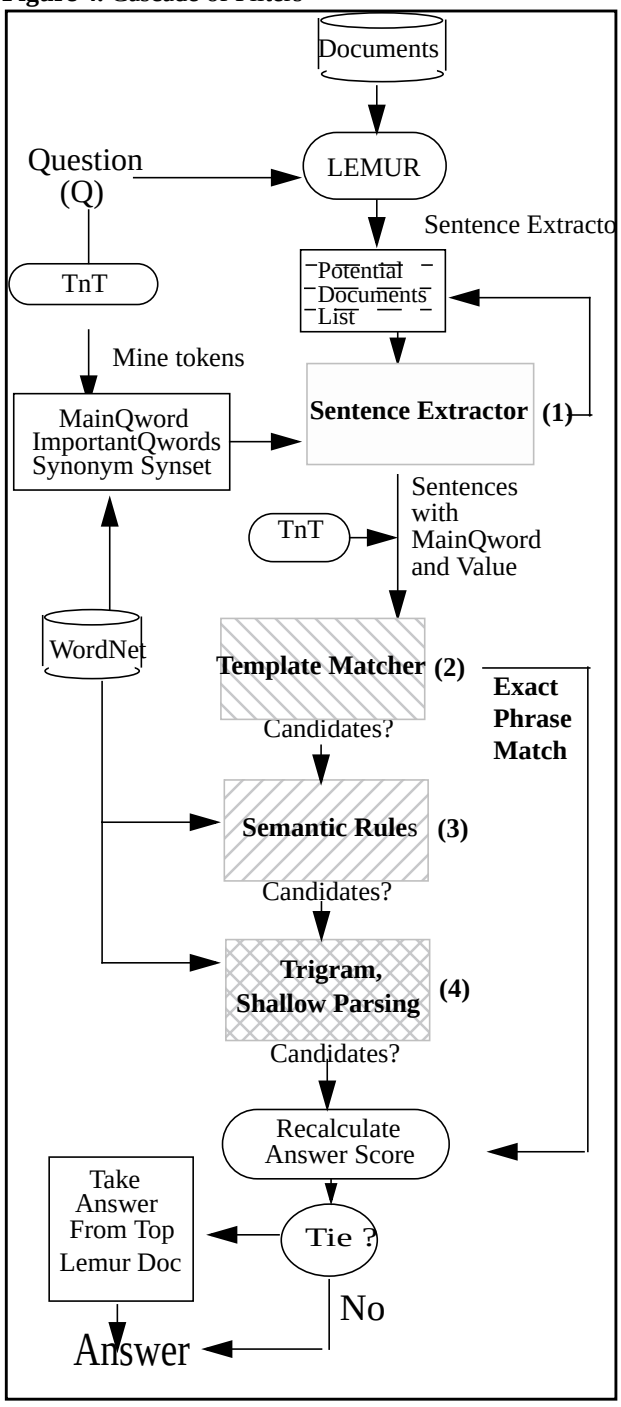
These changes resulted in significant improvements for answering "what" questions and, in some cases, yielded improvements in other categories as well. We were pleased by the gains delivered by these enhancements which affected not only the factoid-answering capability of the system but, even more so, the list-answering. In our tests on past-year development sets, these enhancements appeared to provide MRRs and F-scores of about twice as large as were had using the previous year's QACTIS system.

2.4.2. Cascade of Filters Strategy

When the question type was determined to be either a "How many" or "Other" type question, QACTIS uses a Cascade of Filters approach (CFA) to answer the questions. CFA is faster than the Knowledge-Graph Induction algorithm, answers "how many" questions 10-20% more accurately, and has internal processes that lend themselves well to answering "other" questions.

Synopsis of TREC-2004's CFA: CFA applies different filters to identify potential answers and to eliminate those that are suspect. These filters, which are depicted in Figure 4, were described in detail in last year's proceedings. However, for clarity, these filters consist of:
(1) a *sentence extractor filter,* which identifies potential answer sentences from the top *N* returned IR documents;
(2) a *template matcher filter*, which use regular expressions to find exact or near-exact phrase matches;
(3) a *semantic rules filter*, which attempts to use semantic rules to support or dispose of a candidate answers; and
(4) a *trigram and shallow parsing filter* which attempts to find syntactic similarities between questions and answers.

Figure 4. Cascade of Filters



Internal Modifications of the CFA for 2005: There were only a few changes to CFA over that of the previous year in terms of the answering of "how many" style questions.

Most of these could be relegated to code fixes or minor modifications. On the other hand, there were a number of improvements that were made in "other" question answering. One of these fixes, as mentioned before, was requiring a more lenient second pass at question-answering in the event that the first pass returned no answer. Another, more significant improvement was that last year's "other" answer relied on the direct object as a means of identifying the key query words. This year's version instead used noun phrases from the topics in the first pass, and direct objects only in the second pass. Also, the system used only the top 30 answers from the top 15 documents. Both last year's and this year's versions of "other" answers were used in the competition since last year's had been quite successful and we wanted to continue with that capability while testing this new capability.

3. SYSTEM EVALUATIONS

3.1. Description of Results

In TREC-2005, we submitted three very similar runs which have been, to some degree, described already. The first of these runs made use of exactness and NIL filtering of factoid questions, returned the top 15 candidate answers for lists, and used the new "other" processing just desscribed (with the top 30 answers returned from the top 15 documents). Our second run made use of last year's "other" answerer (returning the top 50 answers from the top 30 documents) and used NIL filtering but not exactness filtering. The third system used the new "other" answer, but no forms of factoid filter. The results of these runs are detailed in Table 1. To our surprise, none of these variations provided significantly different results. We did recognize that there had been a parsing bug in our system for factoid and list answering in the 1st and 3rd runs, but it appears that the bug had little effect. Perhaps the biggest surprise is the "Other" score for systems 1 and 3 which were exactly the same but with different scores.

Table 1: TREC 2005 Performance

Strategy	Factoid	List	Other	All
Exactness & NIL Filter+ New Definer (+parse bug) [#1]	.254	.105	.239	.219
NIL Filter+ modified Original Definer [#2]	.257	.103	.241	.221
New Definer (+parse bug) [#3]	.257	.105	.248	**.222**

3.2. Introspection About Results

In some ways, the scores we observed were lower than we had expected. We had made many serious improvements to the system and expected at least a 5% improvement across the board over last year's system. However, the overall system had only a very slight gain over last year's system of about 1%. We had hoped to see a gain of 10-15% absolute in factoids and about a 1-2 point higher F-score for lists. Our other score was only about half of what we had expected. Part of these drops in performance are explainable. For instance, there was the parsing bug just mentioned. There were also several list and factoid questions that return "NIL" because the system did not know what kind of answer if should return for a question of the form "Give ..." There also appear to be some drops in performance due to the subjectivity in evaluation. We identified a number of questions where the correct answer was returned and the assessors without doubt should have marked it as correct. Nonetheless, we expect that this subjectivity would affect all TREC participants equally and would result in somwhat depressed scores overall. In studying the median and maximum scores between last year and this year, it would appear that there was about 10% relative performance decay across the board due.

On the other hand, we were very pleased from some of the results. Our "other" answerer, despite having a much lower score over those of last year, produced the highest or tied for the highest scoring result. Although there is still significant room to produce better scores in the future, it would appear that it is performing at the current state of the art.

4 FUTURE DIRECTIONS

The future of QACTIS still holds a direction of multilingual and multimedia question-answering as a primary goal. We will, nonetheless, participate in the English newswire TREC next year to validate any improvements that our system makes in the next yeatr. Our focus on textual QA for the next year will be on enhancing the induced graph with induced auxiliary information. In particular, we are planning to concentrate on making use of hypernym induction and adding inferencing to our system while improving the resolution of anaphora and cataphora. At the point in which we think the induced graph and searches thereon are at their optimum, our directions will then change to focus exclusively on multilingual, multimedia data.

5 ACKNOWLEDGMENTS

The authors would like to thank John Prange and the AQUAINT program for providing funding for portions of the work involved in this effort.

6 REFERENCES

[1] Schone, P., McNamee, P., Morris, G., Ciany, G., Lewis, S. "Searching Conversational Telephone Speech in Any of the World's Languages," *Interna-*

tional Conference on Intelligence Analysis. McLean, VA. 2005, https://analysis.mitre.org/proceedings/Final_Papers_FilesIA2005_154_camera_ready_paper.pdf

[2] Schone, P., Ciany, G., McNamee, P., Kulman, A., Bassi, T. , "Question Answering with QACTIS at TREC 2004" *The 13th Text Retrieval Conference (TREC-2004*), Gaithersburg, MD. NIST Special Publication 500-261,2004.

[3] The LEMUR System. URL: http://www-2.cs.cmu.edu/~lemur

[4] Mayfield, J., McNamee, P. "JHU/APL at TREC 2005: QA Retrieval and Robust Tracks," TREC-2005, Gaithersburg, MD, 2005. To appear.

[5] Miller G. A., Beckwith R., Fellbaum C., Gross D., and Miller K. J. "WordNet: An online lexical database." *International Journal of Lexicography* 3(4): 235-244, 1990.

[6] P. Schone, J. Townsend, C. Olano, T.H. Crystal. "Text Retrieval via Semantic Forests." TREC-6, Gaithersburg, MD. *NIST Special Publication 500-240*, pp. 761-773, 1997

[7] www.wikipedia.org

[8] National Geospatial-Intelligence Agency, http://earth-info.nima.mil/gns/html/cntry_files.html

[9] US Geological Survey, http://geonames.usgs.gov/stategaz

[10] Snow, R., Jurafsky, D., and Ng, A.Y., "Learning syntactic patterns for automatic hypernym discovery". *NIPS 2004*.

[11] E. Charniak. "A maximum-entropy inspired parser." *Technical Report CS-99-12*, Brown University, 1999.

[12] D. Bikel, R. Schwartz, R. Weischedel, "An Algorithm that Learns What's in a Name," *Machine Learning*, 1999