IBM's Statistical Question Answering System

Abraham Ittycheriah, Martin Franz, Wei-Jing Zhu, Adwait Ratnaparkhi
P.O.Box 218,
Yorktown Heights, NY 10598
{abei,franzm,wjzhu,adwaitr}@watson.ibm.com

Richard J. Mammone
Dept. of Electrical Engineering, Rutgers University,
Piscataway, NJ 08854
mammone@caip.rutgers.edu

Abstract

We describe the IBM Statistical Question Answering for TREC-9 system in detail and look at several examples and errors. The system is an application of maximum entropy classification for question/answer type prediction and named entity marking. We describe our system for information retrieval which in the first step did document retrieval from a local encyclopedia, and in the second step performed an expansion of the query words and finally did passage retrieval from the TREC collection. We will also discuss the answer selection algorithm which determines the best sentence given both the question and the occurrence of a phrase belonging to the answer class desired by the question. Results at the 250 byte and 50 byte levels for the overall system as well as results on each subcomponent are presented.

1 System Description

Systems that perform question answering automatically by computer have been around for some time as described by (Green et al., 1963). Only recently though have systems been developed to handle huge databases and a slightly richer set of questions. The types of questions that can be dealt with today are restricted to be short answer fact based questions. In TREC-8, a number of sites participated in the first question-answering evaluation (Voorhees and Tice, 1999) and the best systems identified four major subcomponents:

- Question/Answer Type Classification
- Query expansion/Information Retrieval
- Named Entity Marking
- Answer Selection

Our system architecture for this year was built around these four major components as shown in Fig. 1. Here, the question is input and classified as asking for an answer whose category is one of the named entity classes to be described below. Additionally, the question is presented to the information retrieval (IR) engine for query expansion and document retrieval. This engine, given the query, looks at the database of documents and outputs the best documents or passages annotated with the named entities. The final stage is to select the exact answer, given the information about the answer class and the top scoring passages. Minimizing various distance metrics applied over phrases or windows of text results in the best scoring section that has a phrase belonging to answer class. This then represents the best scoring answer.

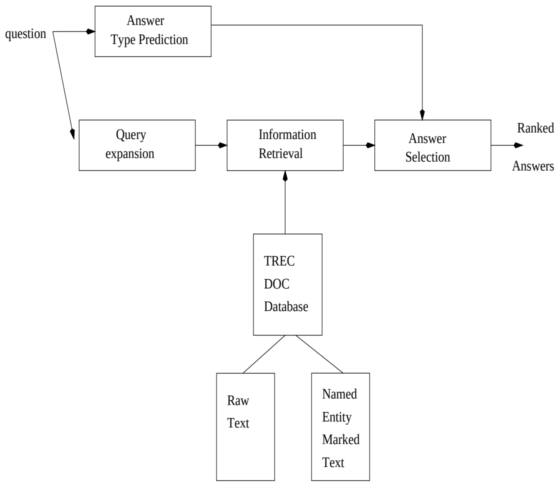

Figure 1: Question Answering Architecture

Maximum entropy modelling is described in (Della Pietra et al., 1995; Berger et al., 1996). Methods of feature selection is further described in (Berger and Printz, 1998). We will not discuss the mathematical details of the algorithm here, instead we will only show the features that are used in such a model.

We will now describe each sub-component in greater detail.

2 Answer Type Classification

In answer type classification the problem is to label a question with the label of the named entity that the question seeks. Our labels are the standard MUC (Chinchor, 1997) categories with the addition of PHRASE which is a catch all for answers not of the standard categories. In addition we had a REASON category which was tied to WHY questions. Processing of REASON and PHRASE is the same in our system, interpreting it as desiring a clause which had a noun phrase embedded in it.

A maximum entropy model of the process was trained on a corpus of questions that has been annontated with the above mentioned categories. We created 1900 questions by presenting a human subject a document selected at random and having read a portion of the document, a question was phrased, the answer and document number noted in addition. We also used 1400 questions from a trivia database (Hallmarks, 1999) annotated in a similar manner.

In the experiments we used the following types of features shown in Table 2. Each feature type expands on the feature above it. The "Expanded Hierarchy" feature uses WordNet (Miller, 1990) to expand words from a question word upto and including the first noun cluster. The "Mark Question Word" feature identifies the question words and labels them as occuring in the beginning of a question (bqw), in the middle (mqw) of a question or at the end of a question (eqw).

The features of the maximum entropy model were n-grams of words (required to be adjacent) and bag of words where position is not important. The performance of the algorithm is shown in Fig. 2. Each feature type adds to the accuracy of the system and choosing 700 features achieves the best accuracy (9.05%) on a held out subset of the data.

A peculiar feature of the architecture is that improvements in answer type prediction do not correlate directly with improvements in the overall score of the system. The reason is that parallel improvements must be made in the named entity marking as well as answer selection in order to realize them in the overall system.

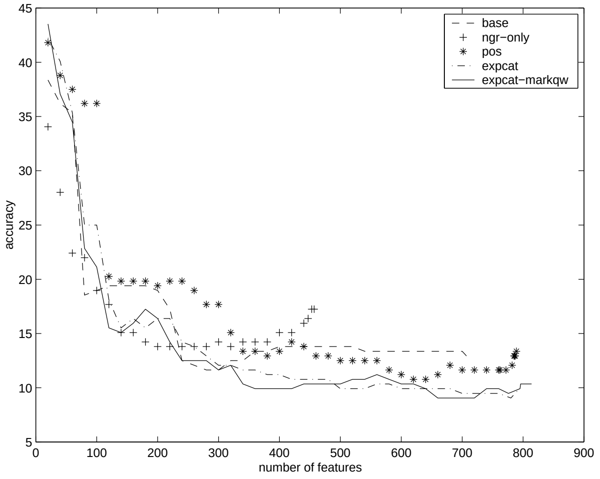

Figure 2: Answer Type Prediction Performance

3 Information Retrieval

The purpose of the information retrieval module is to search the database to select passages of text, containing information relevant to the query.

Our information retrieval subsystem uses a two-pass approach. In the first pass, we searched an encyclopedia database. The highest scoring passages were then used to create expanded queries, applied in the second pass scoring of the TREC passages. We used data pre-processing and relevance scoring techniques similar to the ones we applied in our TREC Ad-Hoc, SDR and CLIR participations (Franz and Roukos, 1998), (Franz et al., 1999).

Relevance scoring was based on morph unigram and bigram features, extracted from the text data using a decision tree based tokenizer, part-of-speech tagger (Merialdo, 1990) and a morphological analyzer.

In the first pass, we used a modification of the Okapi formula (Robertson et al., 1995), described in (Franz et al., 1999) to score passages extracted from the encyclopedia documents. We converted the encyclopedia articles into 82277 overlapping passages, each containing approximately 100 non-stop words. Based on the first pass passage ranking, we constructed expanded queries using the local context analysis (LCA) technique (Xu and Croft, 1996), modified as described in (Franz et al., 1999). In the second pass scoring, we used the expanded queries to score 2632807 passages based on the TREC-9 Q&A corpus. The passages were selected to contained approximately 200 non-stop words.

Table 2 summarizes the information retrieval results on the 146 development test set questions described below. The performance is measured by the

Unigrams	What year did World War II start?
Morphed,Part-Of-Speech	what{WP} year{NN} do{VBD} World{NP} War{NP} II{NP} start{NN}
Bigrams	what{wp} what{wp}_year{nn} what{wp}_do{vbd} what{wp}_world{np} ...
Expanded Hierarchy	what{WP} year time_period measure abstraction year{NN} do{VBD} ...
Mark Question Word	what_bqw year time_period measure abstraction year{NN} do{VBD} ...

Table 1: Features used in the answer classification experiments

	MRR
pass1, TREC	0.4605
pass2, TREC	0.4824
pass2, encyclopedia	0.5031

Table 2: Retrieval results.

Mean Reciprocal Rank (MRR) (Voorhees and Tice, 1999) of the highest ranking passage containing the answer string among the top five passages. The first line of the table shows the result of first pass scoring using the TREC-9 Q&A database. The second line contains the result obtained with queries expanded using the TREC database. The last line of the table shows the result corresponding to the system applied in our official submission, with queries expanded using the encyclopedia database.

4 Named Entity Annotation

Named entity annotation is a markup of the text with the class information. As mentioned above, our classes correspond to the MUC classes due to the availability of training data for these classes. We used the text corpora available from the LDC to train the maximum entropy model.

Windows of +/- 2 words, morphs, part-of-speech tags and flags raised by pattern grammars for DATE, MONEY, CARD, MEASURE, PERCENT, TIME, DURATION classes, along with the two previous tags are created for each word. The window for predicting the tag(0) is shown in Table 3. Each stream has a fixed vocabulary and n-grams from this vocabulary form the features of the maximum entropy model. The training data is arranged to indicate a special category for beginning each named entity, for example BeginPERSON to find the boundaries of the named entity.

The system explores multiple NE hypotheses in parallel and keeps only those with high probability and proceeds with a beam-search algorithm to find the most likely path for the whole sentence. The performance of the named entity detector is comparable to the performance cited in (Borthwick et al., 1998) when training the maximum entropy algorithm on only annotated data. We omit the results here in the consideration of space, but note that in the analysis of the question answering system below only 4 out of 64 errors are attributed directly to the named entity marking for the 250 byte system.

<table>
<tr><td>Words</td><td>w(-) w(-) | w(0) | w(+) w(+)</td></tr>
<tr><td>Morphs</td><td>m(-) m(-) | m(0) | m(+) m(+)</td></tr>
<tr><td>Part-of-Speech</td><td>p(-2) p(-1) | p(0) | p(1) p(2)</td></tr>
<tr><td>Grammar Flags</td><td>f(-2) f(-1) | f(0) | f(1) f(2)</td></tr>
<tr><td>Previous Tags</td><td>t(-2).t(-1) t(-1) |</td></tr>
</table>

Table 3: Features used in the named entity model for predicting tag(0).

5 Answer Selection

We receive in this module the question, the class of the answer that the question seeks and a ranked set of passages (70) annotated with the MUC classes. The optimal sentence that answers the question is now sought. The TREC length constraints of 250 byte and 50 byte are then applied on the sentence.

The algorithm used in this module is listed here:

1. Each retrieved passage is split into sentences.

2. A window is formed around each sentence (window size is 3 sentences)

3. The following distances are computed: Matching Words, Thesaurus Match, Mis-Match Words, Dispersion, and Cluster Words. These are defined below.

4. The location or absence of the desired entities is noted in the score.

5. Each of these distances are weighted, the sentences ranked and the top 5 sentence are then output.

The definition of the various distances are

Matching Words The TFIDF sum of the number of words that matched identically in the morphed space. (+)

Thesaurus Match The TFIDF sum of the number of words that matched using a thesaurus match using WordNet synonyms ((Miller, 1990)). (+)

Mis-Match Words The TFIDF sum of the number of question content words that did not match in this answer. (-)

Dispersion The number of words in the candidate sentence that occur between matching question words. (-)

Cluster Words The number of words in the candidate sentence that occurred adjacently in both the question and answer candidate. (+)

Each distance has a weight applied and the corresponding sign shown above attached to it. The score for an answer is the sum of the distances and the top 5 sentences are then output.

To select the 250 or 50 byte answer chunk from these sentences, the system identified the longest mismatched pieces between the answer and the question. It then analyzed the answer and the question to find where the center of the match was and using a subject-verb-object assumption of the sentence, it took the question as either desiring the subject or object whichever had the least matches with the question.

Answer selection as done above used mostly heuristic distance metrics to seek an answer. Future work by the authors will show how to treat these distance metrics as features and to develop a statistical model for answer selection for an open domain.

6 Development Set Analysis

We wanted to maintain the TREC-9 database as a test set, but in order to do some post-evaluation analysis, we chose a subset of the questions as a development set for next year. There were two classes of questions in this years evaluation: questions that had only one phrasing and questions that had more than one phrasing (rephrased). We wanted 20% of questions of each class in the development test. The exact list of questions we used for our TREC-9 development test set are shown in Table 4. The variant questions we chose are shown in italics, and we added every seventh question skipping the ones in the above class to yield the 146 questions. A set of answer patterns was developed for the set using the judgements file provided by NIST.

201	203	209	210	217	220	224	231	238	242
245	252	253	259	264	266	273	275	280	286
287	294	297	301	308	315	319	322	329	330
336	341	343	350	352	357	363	364	371	374
378	385	392	393	399	411	412	413	420	434
453	454	456	458	462	469	473	476	483	484
490	495	497	504	506	511	517	518	525	528
532	539	546	550	553	560	561	567	572	574
581	583	588	594	595	602	605	609	616	623
627	630	637	638	644	649	651	658	660	665
671	672	679	682	686	693	700	*711*	*712*	*713*
714	*715*	*716*	*717*	*718*	*719*	*720*	*721*	*722*	*723*
724	*725*	*726*	*727*	*728*	*729*	*730*	*731*	*732*	*733*
734	*805*	*806*	*807*	*828*	*829*	*830*	*831*	*832*	*833*
834	*839*	*840*	*841*	*842*	*843*				

Table 4: Question numbers chosen for the TREC-9 development set.

System	TREC9 results	DEV WB expansion	DEV TREC expansion
250 byte	0.457	0.437	0.417
50 byte	0.290	0.287	0.266

Table 5: MRR for TREC-9 and the chosen dev set

The MRR for the entire system for the 250 byte system and the 50 byte system is shown in Table 5.

Analysis of the components are shown in Table 6. An error is attributed to a component if it is the first component that caused the failure working left to right in our system architecture. Fixing this error though need not correct the final answer as it may invoke an error in a subsequent system. Answer selection is still seen to be the major cause of problems in our question answering system.

Component	Number of Errors (Error rate)	
	250 byte	50 byte
Answer Type	5 (3.4%)	7 (4.8%)
IR	19 (13%)	19 (13%)
NE	4 (2.7%)	5 (3.4%)
Answer Selection	36 (24.7%)	52 (35.6%)
System	64(43.8%)	83(56.8%)

Table 6: Component error rate for the TREC9 dev set for 250 byte system

Q&A rank	IR rank						Total
	1	2	3	4	5	5+	
1	29	9	5	3	2	5	53
2	10	2	1	0	0	0	13
3	2	2	1	0	1	0	6
4	1	1	0	1	1	2	6
5	2	1	0	0	1	0	4
5+	13	7	2	1	1	40	64
Total	57	22	9	5	6	47	146

Table 7: Rank transition matrix, IR ws Q&A, 250 bytes

Q&A rank	IR rank						Total
	1	2	3	4	5	5+	
1	20	5	2	1	0	3	31
2	5	2	1	0	0	1	9
3	6	2	1	1	1	0	11
4	3	1	0	0	1	1	6
5	2	1	1	0	1	1	6
5+	21	11	4	3	3	41	83
Total	57	22	9	5	6	47	146

Table 8: Rank transition matrix, IR ws Q&A, 50 bytes

Another viewpoint is to see the effect of the system on the IR ranking results. This is shown below in Figure 3. Finding the 250 bytes from a passage that is of typical length 2.4K bytes shows some degradation, but further finding the 50 byte answer has considerable degradation. In Tables 7 and 8 we show the transition matrix for the rank from IR to the Q&A system. Note that there are significant transitions between the IR rank and the Q&A rank, but that inspection of the final result in Figure 3 shows that overall system performance is similar to the performance of IR for the 250 byte system and degraded at 50 bytes. This we believe points to the possibility of making more improvements in answer selection by reranking the results.

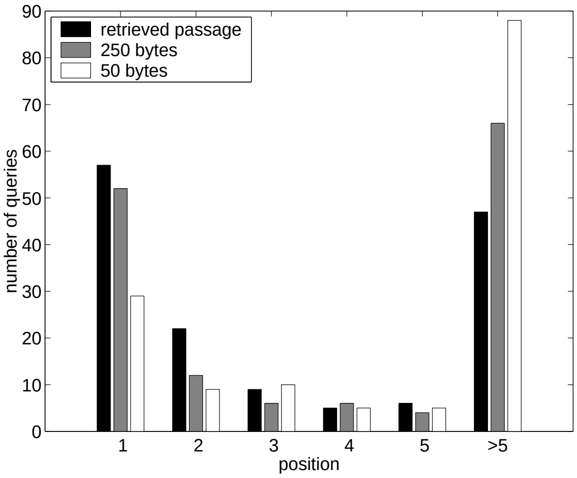

Figure 3: Development Set Performance

7 Conclusion

We presented above our architecture and a component wise evaluation of the system in the question answering problem. This was our first year of developing this system and having performed above the mean we believe that much more can be done in future evaluations. Our current work is to utilize maximum entropy features in the answer selection process which will render the system completely trainable from examples.

8 Acknowledgement

The authors thank Salim Roukos, Kishore Papineni and Todd Ward for making this work possible and helping us with their enormous expertise.

References

Adam Berger and Harry Printz. 1998. A comparison of criteria for maximum entropy/minimum divergence feature selection. *Proceedings of the Third Conference on Empirical Methods in Natural Language Processing*, pages 97–106, June.

Adam L. Berger, Vincent Della Pietra, and Stephen Della Pietra. 1996. A maximum entropy approach to natural language processing. *Computational Linguistics*, 22(1).

A. Borthwick, J. Sterling, E. Agichtein, and R. Grishman. 1998. Exploiting diverse knowledge sources via maximum entropy in named entity recognition. *Proceedings of the COLING-ACL 98, Sixth Workshop on Very Large Corpora.*

Nancy Chinchor. 1997. Muc-7 named entity task definition. *Proceedings of MUC-7.*

Stephen Della Pietra, Vincent Della Pietra, and John Lafferty. 1995. Inducing features of random fields. *Technical Report CMU-CS-95-144*, May.

M. Franz and S. Roukos. 1998. Trec-6 ad-hoc retrieval. In E.M. Voorhees and D.K. Harman, editors, *Proceedings of the Sixth Text REtrieval Conference (TREC-6).* NIST Special Publication 500-240.

M. Franz, J. S. McCarley, and S. Roukos. 1999. Adhoc and multilingual information retrieval at ibm. In E.M. Voorhees and D.K. Harman, editors, *Proceedings of the Seventh Text REtrieval Conference (TREC-7)*. NIST Special Publication 500-242.

B.F. Green, A.K. Wolf, C. Chomsky, and L.K. Baseball. 1963. An automatic question answerer. *Computers and Thought*, pages 207–216.

Academic Hallmarks. 1999. Knowledge master. *http://www.greatauk.com.*

B. Merialdo. 1990. Tagging text with a probabilistic model. In *Proceedings of the IBM Natural Language ITL*, pages 161–172.

G. Miller. 1990. Wordnet: An on-line lexical database. *International Journal of Lexicography*, 3(4).

S.E. Robertson, S. Walker, S. Jones, M. M. Hancock-Beaulieu, and M. Gatford. 1995. Okapi at trec-3. In D.K. Harman, editor, *Proceedings of the Third Text REtrieval Conference (TREC-3)*. NIST Special Publication 500-225.

Ellen M. Voorhees and Dawn M. Tice. 1999. The trec-8 question answering track evaluation. *TREC-8 Proceedings*, pages 41–63.

Jinxi Xu and W. Bruce Croft. 1996. Query expansion using local and global document analysis. In *Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval.*