The Pronto QA system at TREC-2007: harvesting hyponyms, using nominalisation patterns, and computing answer cardinality

Johan Bos[1], **James R. Curran**[2], **Edoardo Guzzetti**[1]

[1]Dept. of Computer Science
University of Rome "La Sapienza"
bos@di.uniroma1.it

[2]School of IT
University of Sydney
james@it.usyd.edu.au

Abstract

The backbone of the Pronto QA system is linguistically-principled: Combinatory Categorial Grammar is used to generate syntactic analyses of questions and potential answer snippets, and Discourse Representation Theory is employed as semantic formalism to match the meanings of questions and answers. The key idea of the Pronto system is to use semantics to prune answer candidates, thereby exploiting lexical resources such as WordNet and NomLex to facilitate the selection of answers. The system performed well at TREC-2007 on *factoid*-questions with an answer accuracy of 22%, a score higher than the median accuracy score of all participating systems.

1 Introduction

The QA evaluation exercise at TREC consists in automatically finding answers for a collection of questions arranged by different topics, or *targets* in TREC parlance. Questions can be either *factoid*-questions, asking for a unique short answer, or *list*-questions, asking for a set of answers. Each series of questions ends with an *other*-question, which is a request to provide all relevant information about the target which was not already asked in the previous questions. An example of a target and its questions is shown in Figure 1.

The answers must be presented with their sources. For TREC-2007, the relevant document collections were Aquaint-2 and Blog06. Aquaint-2 is collection of almost one million newswire articles (written in English) dating from 2004–2006. Blog06 is a set of homepage documents from late 2005 and early 2006. A response is evaluated as correct only if it exactly answers the question (in an exhaustive but not overinformative way), if it is the most recent correct answer (i.e., globally correct rather than locally correct), and if it is accompanied by a document ID from one of the two previously mentioned corpora supporting the answer.

This paper contains a description and results of our QA system "Pronto" at TREC-2007. Probably the most interesting aspect of the Pronto system is that it uses a deep linguistic analysis, combining symbolic with statistical approaches, and its use of general background knowledge as found in resources such as WordNet and NomLex. The Pronto system is a reincarnation of the "La Sapienza" system (Bos06), which was in turn inspired by the QED system (LBD$^+$03; ABC$^+$04; ABC$^+$05).

The major modifications with respect to this earlier versions concern mainly question analysis (supporting multiple interpretations of the question), the use of large corpora to find relevant hyponyms, named entity recognition of creative works (Guz07), computing the answer cardinality for *list*-questions, and the resolution of indirect temporal expressions.

2 The Pronto QA system

As with any open-domain QA system, the input of Pronto is a question, and its output a set of answers. In between we have a cascaded architecture of components, consisting of question interpretation (parsing and boxing the question), computing relevant background knowledge, expected answer typing, query construction, document retrieval, answer extraction, and finally answer selection.

2.1 Question Interpretation

Each question in a serie is analysed together with its corresponding target. The steps of processing are: tokenisation, morphological analysis with Morpha (MP01), named entity recognition, part of speech tagging, parsing, and semantic construction.

Two parsers are employed to maximise coverage: the wide-coverage CCG-parser of Clark & Curran (CC04) and the Pronto in-built question parser for CCG. On the basis of the output of the parsers a semantic representation is constructed with the help of Boxer (BCS$^+$04;

TARGET: Rubik's Cube Competitions

244.1	(factoid)	Who invented the Rubik's Cube?
244.2	(factoid)	Who founded the International Rubik's Cube Competition in the United States?
244.3	(factoid)	What was the world record time set in the 2006 competition?
244.4	(factoid)	What was the previous world record time?
244.5	(factoid)	Who is considered to be the fastest Rubik's Cube solver on the planet?
244.6	(list)	Who have set world records in solving Rubik's Cubes?
244.7	(other)	

Figure 1: Example of a TREC-2007 serie of questions for a target.

Bos05). Boxer produces a Discourse Representation Structure (DRS), closely following Discourse Representation Theory (KR93), a formal theory of natural language meaning.

For each question a set of expected answer types is produced. The result of question interpretation and answer type determination is a Question-DRS (Q-DRS for short), or in case of ambiguities, several Q-DRSs. An example of a Q-DRS is shown in Figure 2.

2.2 Computing Background Knowledge

Many questions require additional background information in order to be able to answer it with a high level of confidence. Here we have in mind background knowledge derived from large databases such as WordNet (Fel98) and NomLex (MMY^{+}98). But to use all the knowledge encoded in WordNet or other large resources for each question and potential answer pair would obviously render the overall system inefficient. Instead, Pronto comes with a component that computes all background knowledge for a given question that is expected to be useful in later stages of the processing pipeline.

Put differently, the background knowledge for a question constitutes a list of axioms related to the question. It is gathered from lexical resources on the basis of the non-logical symbols that occur in the semantic representation of the question (the Question-DRS). Currently the following kinds of axioms are used:

- synonyms, plus direct and indirect hyponyms and hyperonyms for nouns and verbs derived from WordNet (Fel98);
- synonyms of names derived from WordNet (Fel98);
- hyponyms for nouns harvested from corpora (Aquaint-1, Aquaint-2, and the web) using lexical patterns using techniques similar as in (Hea92);
- nominalisation rules generated from NomLex (MMY^{+}98);
- specialised knowledge, such as attributes (colours, shapes), and geographical knowledge (continents, states, countries, capitals).

The background knowledge for a question is used for determining the expected answer type, for generating queries in the document retrieval stage, and in answer extraction and selection.

2.3 Answer Cardinality

For each *list*-question the expect number of answers is computed. We refer to this as *answer cardinality*, which denotes a range expressed by an ordered pair of two numbers. The first of these numbers indicates the mininal number of answers expected (the lower bound), the second the maximal number of answers (the upper bound, which is set to 0 if unknown). For instance, the answer cardinality "3–3" indicates that exactly three answers are expected, and "2–0" means that the question requires at least two answers.

For the TREC-2007 exercise, Answer Cardinality was computed by reformulating the *list*-question into a question asking for a number. For instance, considering the question 268.7, "What were the settlements that were evacuated?", with target "Israel evacuation of the Gaza Strip", a new question was generated with the surface form "How many were the settlements that were evacuated?". This question was given to the Pronto system, and the returned answer regarded as answer cardinality. For this specific example, Pronto found the following (correct) answer:

> [XIN_ENG_20050630.0060] Israel is scheduled to evacuate **21** settlements in the Gaza Strip and four in northern West Bank from Aug. 17 in order to " disengage " from conflicts with the Palestinians.

The questions were reformulated using simple rewrite rules with the help of regular expressions. To increase the chance of finding a correct answer, documents of both Aquaint-1 and Aquaint-2 were taken into account in the document retrieval stage. If no reliable answer was found, an answer-cardinality of 12 was returned, a number considered to be a good default based on previous TREC campaigns. If a number higher than 100 was found, answer cardinalty was “corrected down” to 100.

2.4 Document Retrieval

All documents in the Aquaint-2 corpus were preprocessed: the XML was stripped off, sentence boundaries were detected using Punkt (KS06), and all text was tokenised. The documents were then rearranged into smaller documents of two sentences each (taking a sliding window, so each sentence appeared in two mini-documents). These mini-documents were indexed with the Indri information retrieval tools (MC04).

For each query, up to 5,000 mini-documents were retrieved, again with the help of Indri (MC04). At this stage of processing, the aim is high recall at the expense of precision. By selecting a high number of documents, the pool of potential answers can be narrowed down as late as possible in the processing pipeline. Processing a high number of documents is certainly time-consuming, but as there are no important time-constraints in the TREC exercise, this is no big concern and advantage is taken of the situation.

2.5 Document Analysis

With the help of the C&C wide-coverage CCG parser, all retrieved mini-documents are parsed and for each of them a Discourse Representation Structure (DRS) is generated using Boxer. The parser also performs basic named entity recognition for locations, persons, and organisations. In addition, a named entity recogniser for creative works was used too (Guz07), as usually a substantial part of questions required a name of a creative work (book, film, play) as answer (Bos06).

Each mini-document is translated into a single DRS (the so-called A-DRS). A set of DRS normalisation rules are applied in a post-processing step, thereby dealing with active-passive alternations, inferred semantic information, normalisation of temporal expressions, and the disambiguation of noun-noun compounds. The resulting DRS is enriched with information about the original surface word-forms and parts of speech. An example of an A-DRS is shown in Figure 2.

2.6 Answer Extraction

Given the DRS of the question (the Q-DRS), and a set of DRSs of the retrieved documents (the A-DRSs), each A-DRS is matched with the Q-DRS to find a potential answer. This process proceeds as follows: if the A-DRS contains a discourse referent of the expected answer type matching will commence. The process of matching attempts to identify the semantic structure in the Q-DRS with that of the A-DRS. The result is a score between 0 and 1 indicating the amount of semantic material that could be matched. The generated background knowledge for the question (see Section 2.2) is used to assist in the matching.

2.7 Answer Selection

The Answer Extraction component yields a list of answers and a matching score. Answers that are semantically identical are grouped together. This process produces a new list of answers, ranked on matching score and frequency. A simple method of reranking was employed at the TREC-2007 exercise, namely by sorting on the matching score, using the highest answer frequency as tie-breaker.

2.8 Processing *other*-questions

Since *other*-questions do not appear as ordinarily formulated questions, but the QA system expects questions phrased in English as input, they are automatically transformed into definition questions. This is simply done by parsing the question “What did he also do?” and assigning it the answer type DEFINITION with answer cardinality 1–0. The answer extraction component deals with definition questions by finding statements that directly relate the target to numeral expressions, date expressions, adjectives, or proper names.

3 Evaluation

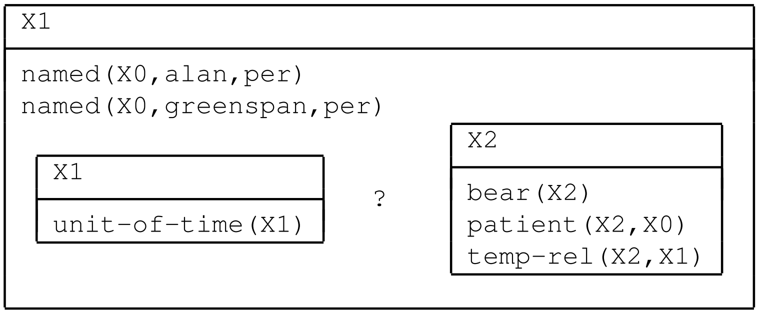
3.1 Experimental Setup

Three runs were submitted, all with different parameters with respect to the treatment of *factoid*, *list*, and *other*-questions. The parameters for *factoid*-questions were the use of hypernyms, the use of hyponyms harvested from large corpora (i.e., not from WordNet), and whether documents from the Blog06 corpus were included in the search or not. The first run for *list*-questions selected the twelve best matching answers, whereas the second and third run used our answer cardinality method (Section 2.3), to select the N-best answers. For *other*-questions, we increased the number of selected answers with each run. Table 1 summarises the runs and the parameters used.

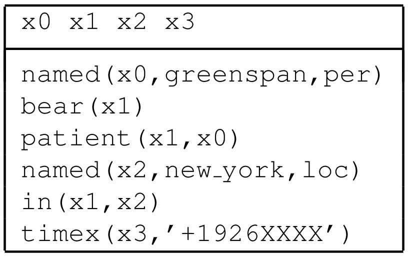
3.2 TREC-2007 Judgements

Factoid questions, as usual, formed the majority of the questions at the TREC-2007 QA evaluation exercise. The results of the Pronto system over 360 factoid questions

Question:	When was Alan Greenspan born?
Target:	Alan Greenspan
ID:	264.1

Q-DRS:	 X1
	named(X0,alan,per)
	named(X0,greenspan,per)
	X1
	unit-of-time(X1)
	?
	X2
	bear(X2)
	patient(X2,X0)
	temp-rel(X2,X1)

Answer Type:	[match:name:year,match:name:day]
Cardinality:	1–1
Query	greenspan
Context:	[AFP_ENG_20060125.0276] Greenspan was born in New York in **1926**

A-DRS:	 x0 x1 x2 x3
	named(x0,greenspan,per)
	bear(x1)
	patient(x1,x0)
	named(x2,new_york,loc)
	in(x1,x2)
	timex(x3,'+1926XXXX')

Answer:	`264.1 pronto07run3 AFP_ENG_20060125.0276 1926`

Figure 2: System input and output for the *factoid*-question 264.1 at TREC-2007.

Table 1: Description of the three runs of Pronto at TREC-2007, for *factoid*, *list* and *other*-questions.

Run	Hypernyms	Extra Hyponyms	Blog06	List	Other
1	yes	no	no	12-best	6-best
2	no	yes	no	N-best	12-best
3	no	yes	yes	N-best	18-best

are listed in Table 2, where U is the number of unsupported (correct but without a supporting document), X the number of inexact, L the number of locally correct (a later document in the Acquint corpus contradicts the answer, so the response is judged locally rather than globally correct), and R the number of correct answers.

Table 2: Results for *factoid*-questions, TREC-2007.

Run	U	X	L	R	Acc.	Len. Acc.
1	3	9	7	70	0.19	0.25
2	3	11	7	75	0.21	0.27
3	5	15	8	80	0.22	0.30
all	529	756	169	3055	0.17	0.25

The last two columns of Table 2 show the accuracy (calculated on the basis of R) and lenient accuracy (calculated on the basis of U+X+L+R). In addition, it shows the summed scores of all participating systems at TREC-2007, a total of 51 runs. As suspected, adding extra hyponyms improves accuracy, whereas adding hypernyms can actually have a negative influence. Our third run showed the best results, which only differed from the second run by inclusion of the Blog06 documents.

Table 2 also shows that, compared with the total of all runs, Pronto produced relatively few unsupported answers, and relatively many locally correct answers. The former is probably due to the deep linguistic analysis for matching answers to questions, and refraining from using the web as an additional corpus. The latter is likely caused by the fact that Pronto doesn't distinguish locally correct from globally correct answers.

There were 85 *list*-questions in total. These are evaluated by calculating the precision and recall for each question and then averaging their corresponding F-scores. For the third run, the Pronto system achieved an average F-score slightly higher than the median of all participating systems (Table 3).

The results of the *other*-questions were slightly disappointing compared to last year's results (Bos06). The per-series results, where the results of factoid, list and other questions are combined into one summarising score, were higher than the medium score of all participating systems.

Table 3: Results: average F-score for *list* question, Pyramid F-score for *other*-questions, and per-series scores at TREC-2007.

Run	List	Other	Series
1	0.09	0.04	0.11
2	0.09	0.06	0.12
3	0.10	0.07	0.13
median	0.09	0.12	0.11
best	0.48	0.33	0.48

3.3 A Closer Look

Here we look at three specific aspects of the Pronto QA system that we believe contributed to its performance at TREC-2007: the use of nominalisation patterns in the background knowledge, harvesting hyponyms from large corpora, and the use of answer cardinality in *list*-questions.

3.3.1 Using NomLex

Although we didn't specifically evaluate the contribution of NomLex (MMY$^+$98), we found several examples where NomLex clearly contributed to finding a correct answer. A case in point is the factoid question in Figure 3. Here the background knowledge generated from the NomLex patterns facilitates the matching between question and answer. As a result, a higher matching score is obtained, and the correct answer is selected.

3.3.2 Using Hyponyms found in Corpora

Hyponyms are probably the most important lexical relation required in question answering. Wordnet (Fel98) contains many hyponyms, but in the majority of cases the hyponym is a common noun, rather than a proper name. For many questions, proper name hyponyms (sometimes also referred to as *instances*) are crucial in selecting appropriate answers. This hypothesis was confirmed in TREC-2007, as is illustrated for a few selected examples in Table 4.

Table 4: Examples of hyponyms found in large corpora, thereby selecting a correctly judged answer.

ID	BK	Source
216.1	$\forall \mathrm{x}(\mathrm{new\_york\_times}(\mathrm{x}) \rightarrow \mathrm{newspaper}(\mathrm{x}))$	Aquaint
220.7	$\forall \mathrm{x}(\mathrm{tiger}(\mathrm{x}) \wedge \mathrm{woods}(\mathrm{x}) \rightarrow \mathrm{star}(\mathrm{x}))$	Web
222.7	$\forall \mathrm{x}(\mathrm{scotchgard}(\mathrm{x}) \rightarrow \mathrm{brand}(\mathrm{x}))$	Web
222.7	$\forall \mathrm{x}(\mathrm{upn}(\mathrm{x}) \rightarrow \mathrm{network}(\mathrm{x}))$	Aquaint
232.6	$\forall \mathrm{x}(\mathrm{independence\_air}(\mathrm{x}) \rightarrow \mathrm{airline}(\mathrm{x}))$	Aquaint
232.6	$\forall \mathrm{x}(\mathrm{jetblue}(\mathrm{x}) \rightarrow \mathrm{airline}(\mathrm{x}))$	Aquaint
255.5	$\forall \mathrm{x}(\mathrm{imperial}(\mathrm{x}) \rightarrow \mathrm{company}(\mathrm{x}))$	Web

Question:	Who founded the International Rubik's Cube Competition in the United States?
Target:	Rubik's Cube Competitions
ID:	244.2
Answer:	Tyson Mao
Context	Tyson Mai, founder of the International Rubik's Cube Competition in the United States, competes in the 3x3x3 Blindfolded Rubik's Cube competition during the International Rubik's Cube Competition in San Francisco on Saturday Jan. 14, 2006.
Source:	BLOG06-20060119-069-0013050325
BK:	$\forall x \forall y(\text{founder}(x) \wedge \text{of}(x,y) \rightarrow \exists e(\text{found}(e) \wedge \text{agent}(e,x) \wedge \text{patient}(e,y)))$

Figure 3: Example of background knowledge generated by NomLex

3.3.3 Computing Answer Cardinality

The computation of answer cardinality was put into action in the second and third run (Table 1) but hardly seemed to make a difference in the overall results for *list*-questions, as Table 3 suggests. A good reason to have a closer look as to what happened. What we did is the following: we compared the number of answers returned by Pronto with the number of known answers as reported in the official judgement files distributed by the organisers of TREC.

It turned out that the baseline, guessing an answer cardinalty of 12 or less when fewer answers were found, had a precision of 5/48. The method described in Section 2.3 reached a precision of 7/48. This is of course only a rough indication of success and could just be a coincidence. A better way to measure accuracy of answer cardinality given AC$_k$, the known number of answers, and ACc, the computed answer accuracy, is:

$$\text{Acc} = 1 - \frac{|AC_K - AC_C|}{AC_K + AC_C}$$

Using this equation, the baseline yielded an average accuracy of 0.62, whereas our experimental method reached a lower average accuracy of 0.57. In other words, our novel method for computing answer cardinality failed to beat the baseline. It is interesting to note that the best default answer cardinality for the *list*-questions of TREC-2007 turned out to be 7, reaching an average accuracy of 0.72 (or 0.63 when taking 7 as an upper bound of selected answers).

4 Conclusion

The Pronto QA system is based on a deep linguistic analysis of question and potential answers contexts and uses semantics to narrow down the number of answer candidates. Compared to other QA systems at TREC-2007, Pronto performed above par for *factoid* and *list*-questions. We have shown that the use of hyponyms obtained from large corpora, and patterns translated from NomLex increase the performance of the system. We tried to find a simple but effective method for computing answer cardinality when dealing with *list*-questions, but it turns out not to be straightforward to beat a baseline choosing the average number of answers. To compare different approaches to answer cardinality, we introduced a new metric for measuring the accuracy of answer cardinality for *list*-questions.

Acknowledgements

Johan Bos is supported by a "Rientro dei Cervelli" grant (Italian Ministry for Research).

References

K. Ahn, J. Bos, S. Clark, J.R. Curran, T. Dalmas, J.L. Leidner, M.B. Smillie, and B. Webber. Question Answering with QED and Wee at TREC 2004. In Voorhees and Buckland, editors, *The Thirteenth Text Retrieval Conference, TREC-2004*, Gaithersburg, MD, 2004.

K. Ahn, J. Bos, J.R. Curran, D. Kor, M. Nissim, and B. Webber. Question Answering with QED at TREC 2005. In Voorhees and Buckland, editors, *The Fourteenth Text Retrieval Conference, TREC-2005*, Gaithersburg, MD, 2005.

J. Bos, S. Clark, M. Steedman, J.R. Curran, and Hockenmaier J. Wide-Coverage Semantic Representations from a CCG Parser. In *Proceedings of the 20th International Conference on Computational Linguistics (COLING '04)*, Geneva, Switzerland, 2004.

Johan Bos. Towards wide-coverage semantic interpretation. In *Proceedings of Sixth International Workshop on Computational Semantics IWCS-6*, pages 42–53, 2005.

Johan Bos. The "La Sapienza" Question Answering System at TREC 2006. In Voorhees et al., editor, *The Fifteenth Text REtrieval Conference, TREC-2006*, Gaithersburg, MD, 2006.

S. Clark and J.R. Curran. Parsing the WSJ using CCG and Log-Linear Models. In *Proceedings of the 42nd*

Annual Meeting of the Association for Computational Linguistics (ACL '04), Barcelona, Spain, 2004.

C. Fellbaum, editor. *WordNet. An Electronic Lexical Database*. The MIT Press, 1998.

E. Guzzetti. Riconoscimento automatico di nomi propri: un'applicazione al caso dei titoli cinematografici. Master's thesis, University of Rome "La Sapienza", 2007.

M. Hearst. Automatic acquisition of hyponyms from large text corpora. In *Proceedings of the Fourteenth International Conference on Computational Linguistics*, Nantes, France, 1992.

H. Kamp and U. Reyle. *From Discourse to Logic; An Introduction to Modeltheoretic Semantics of Natural Language, Formal Logic and DRT*. Kluwer, Dordrecht, 1993.

Tibor Kiss and Jan Strunk. Unsupervised multilingual sentence boundary detection. *Computational Linguistics*, 32(4):485–525, 2006.

J. L. Leidner, J. Bos, T. Dalmas, J.R. Curran, S. Clark, C.J. Bannard, M. Steedman, and B. Webber. The QED Open-Domain Answer Retrieval System for TREC 2003. In Voorhees and Buckland, editors, *The Twelfth Text Retrieval Conference, TREC-2004*, Gaithersburg, MD, 2003.

D. Metzler and W.B. Croft. Combining the Language Model and Inference Network Approaches to Retrieval. *Information Processing and Management*, 40(5):735–750, 2004.

A. Meyers, C. Macleod, R. Yangarber, R. Grishman, L. Barrett, and R. Reeves. Using nomlex to produce nominalization patterns for information extraction. In *Coling-ACL98 workshop Proceedings, The Computational Treatment of Nominals*, Montreal, Canada, 1998.

Carroll J. Minnen, G. and D. Pearce. Applied morphological processing of english. *Natural Language Engineering*, 7(3):207–223, 2001.