The Use of External Knowledge in Factoid QA

Eduard Hovy, Ulf Hermjakob, Chin-Yew Lin

Information Sciences Institute
University of Southern California
4676 Admiralty Way
Marina del Rey, CA 90292-6695
tel: 310-448-8731
fax: 310-823-6714
email: {hovy,ulf,cyl}@isi.edu

Abstract

This paper describes recent development in the Webclopedia QA system, focusing on the use of knowledge resources such as WordNet and a QA typology to improve the basic operations of candidate answer retrieval, ranking, and answer matching.

1. Introduction

The Webclopedia factoid QA system increasingly makes use of syntactic and semantic (world) knowledge to improve the accuracy of its results. Previous TREC QA evaluations made clear the need for using such external knowledge to improve answers. For example, for definition-type questions such as

Q: what is bandwidth?

the system uses WordNet to extract words used in the term definitions before searching for definitions in the answer corpus, and boosts candidate answer scores appropriately. Such definitional WordNet glosses have helped definition answers (10% for definition questions, which translates to about 2% overall score in the TREC-10 QA evaluation, given that as many as a little over 100 out of 500 TREC-10 questions were definition questions).

This knowledge is of one of two principal types: generic knowledge about language, and knowledge about the world. After outlining the general system architecture, this paper describes the use of knowledge to improve the purity of phase 1 of the process (retrieval, segmenting, and ranking candidate segments), and to improve the results of phase 2 (parsing, matching, and ranking answers).

Webclopedia adopts the by now more or less standard QA system architecture, namely question analysis, document / passage retrieval, passage analysis for matching against the question, and ranking of results. Its architecture (Figure 1) contains the following modules, which are described in more detail in (Hovy et al., 2001; Hovy et al., 2000):

- **Question parsing:** Using BBN's IdentiFinder (Bikel et al., 1999), the CONTEX parser produces a syntactic-semantic analysis of the question and determines the QA type.
- **Query formation**: Single- and multi-word units (content words) are extracted from the analysis, and WordNet synsets are used for query expansion. A series of Boolean queries is formed.
- **IR**: The IR engine MG (Witten et al., 1994) returns the top-ranked *N* documents.
- **Selecting and ranking sentences**: For each document, the most promising $K<<N$ sentences are located and scored using a formula that rewards word and phrase overlap with the question and its expanded query words. Results are ranked.
- **Parsing segments**: CONTEX parses the top-ranked 300 sentences.
- **Pinpointing**: Each candidate answer sentence parse tree is matched against the parse of the question; sometimes also the preceding sentence. As a fallback the window method is used.
- **Ranking of answers**: The candidate answers' scores are compared and the winner(s) are output.

Webclopedia classifies desired answers by their semantic type, using the approx. 140 classes developed in earlier work on the project (Hovy et al., 2000). These types include common semantic classes such as PROPER-PERSON, EMAIL-ADDRESS, LOCATION, and PROPER-ORGANIZATION, but also classes particular to QA such as WHY-FAMOUS, YES:NO, and ABBREVIATION-EXPANSION. They have been taxonomized as the Webclopedia QA Typology, of which an older version can be found at http://www.isi.edu/natural-language/projects/webclopedia/Taxonomy/taxonomy_toplevel.html.

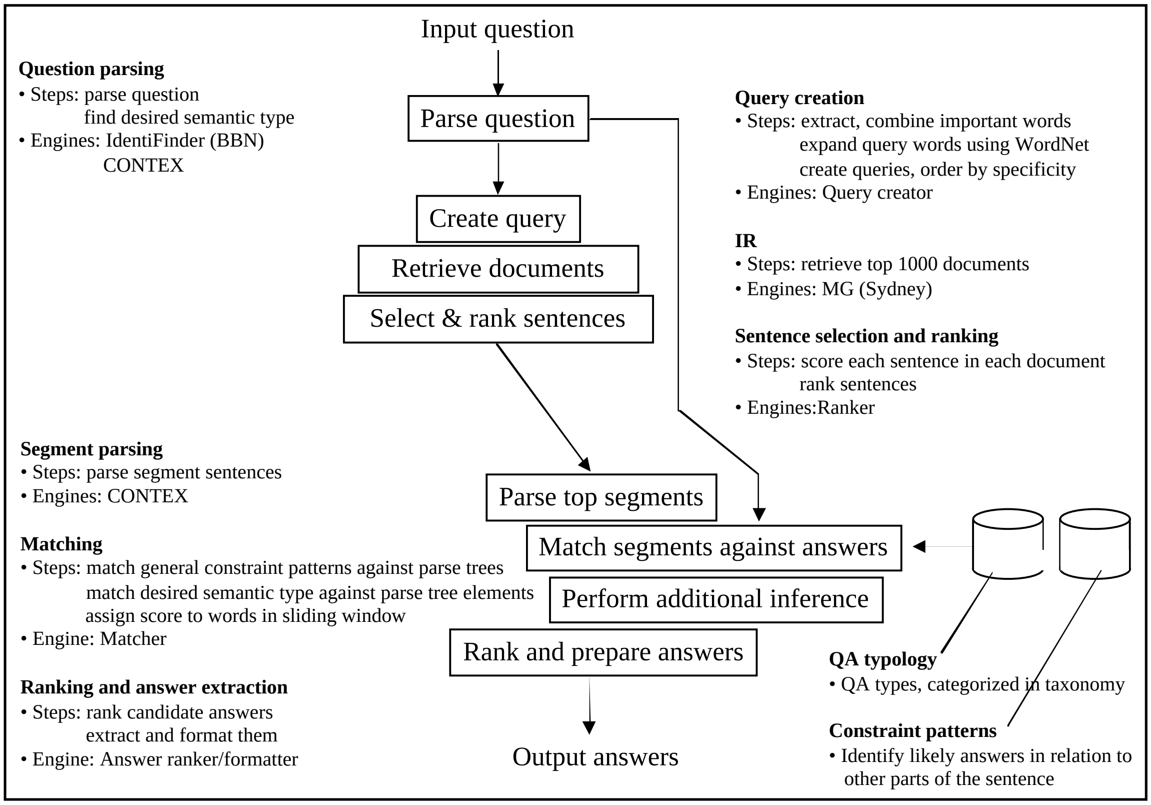

Input question
Question parsing
• Steps: parse question
find desired semantic type
• Engines: IdentiFinder (BBN)
CONTEX
Parse question
Query creation
• Steps: extract, combine important words
expand query words using WordNet
create queries, order by specificity
• Engines: Query creator
Create query
Retrieve documents
Select & rank sentences
IR
• Steps: retrieve top 1000 documents
• Engines: MG (Sydney)
Sentence selection and ranking
• Steps: score each sentence in each document
rank sentences
• Engines:Ranker
Segment parsing
• Steps: parse segment sentences
• Engines: CONTEX
Parse top segments
Matching
• Steps: match general constraint patterns against parse trees
match desired semantic type against parse tree elements
assign score to words in sliding window
• Engine: Matcher
Match segments against answers
Perform additional inference
Rank and prepare answers
QA typology
• QA types, categorized in taxonomy
Ranking and answer extraction
• Steps: rank candidate answers
extract and format them
• Engine: Answer ranker/formatter
Output answers
Constraint patterns
• Identify likely answers in relation to
other parts of the sentence

Figure 1. Webclopedia architecture.

2. Parsing

CONTEX is a deterministic machine-learning based grammar learner/parser that was originally built for MT (Hermjakob, 1997). For English, parses of unseen sentences measured 87.6% labeled precision and 88.4% labeled recall, trained on 2048 sentences from the Penn Treebank. Over the past few years it has been extended to Japanese and Korean (Hermjakob, 2000).

For Webclopedia, CONTEX required two extensions. First, its grammar had to be extended to include question forms. The grammar learner portion of CONTEX was trained on approx. 1150 questions and achieved accuracies of approx. 89% labeled precision and labeled recall (Hermjakob, 2001). Second, the grammar had to be augmented to recognize the semantic type of the desired answer (which we call the *qtarget*). Its semantic type ontology was extended to include currently abourt 140 qtarget types, plus some combined types (Hermjakob, 2001). Beside the qtargets that refer to semantic concepts, qtargets can also refer to part of speech labels (e.g., S-PROPER-NAME) and to constituent roles or slots of parse trees (e.g., [ROLE REASON]). For questions with the Qtargets Q-WHY-FAMOUS, Q-WHY-FAMOUS-PERSON, Q-SYNONYM, and others, the parser also provides *qargs*—information helpful for matching:

Who was Betsy Ross?   QTARGET: Q-WHY-FAMOUS-PERSON   QARGS: ("Betsy Ross")

How is "Pacific Bell" abbreviated? QTARGET: Q-ABBREVIATION QARGS: ("Pacific Bell")

What are geckos?   QTARGET: Q-DEFINITION   QARGS: (("geckos" "gecko") ("animal"))

These qtargets are determined during parsing using approx. 300 hand-written rules.

3. Document Retrieval and Sentence Ranking

Analyzing the Question to Create a Query

We parse input questions using CONTEX (Section 2) to obtain a semantic representation of the questions. For example, we determine that the question “How far is it from Denver to Aspen?” is asking for a distance quantity. The question analysis module identifies noun phrases, nouns, verb phrases, verbs, adjective phrases, and adjectives embedded in the question. These phrases/words are assigned significance scores according to the frequency of their type in our question corpus (a collection of 27,000+ questions and answers), secondarily by their length, and finally by their significance scores, derived from word frequencies in the question corpus.

We remain indebted to BBN for the use of IdentiFinder (Bikel et al., 1999), which isolates proper names in a text and classifies them as person, organization, or location.

Expanding Queries

Query expansion comes from two sources and used in different stages. In the document retrieval stage, the highly relevant question terms (identified by CONTEX) are expanded in order to boost recall, for example going from “Russian” to “Soviet” or from “capital of the United States” to “Washington”. In the sentence ranking stage, we use WordNet 1.6 (Fellbaum, 1998) to match expanded query terms. Although these expanded terms contribute to the final score, their contribution is discounted. This application of expansion strategy aims to achieve high precision and moderate recall.

Retrieving Documents

We use MG (Witten et al., 1994) as our search engine. Although MG is capable of performing ranked query, we only use its Boolean query capability. For the entire TREC-10 test corpus, the size of the inverse index file is about 200 MB and the size of the compressed text database is about 884 MB. The stemming option is turned on. Queries are sent to the MG database, and the retrieved documents are ranked according to their ranking from query analysis. We order queries most specific first, then gradually relax them to more general, until we have retrieved a sufficient number of documents. For example, *(Denver&Aspen)* is sent to the database first. If the number of documents returned is less than a pre-specified threshold, for example, 500, then we retain this set of documents as the basis for further processing, while also submitting the separate queries *(Denver)* and *(Aspen)*.

Ranking Sentences

If the total numbers of sentences contained in the documents returned by MG is N for a given Boolean query, we would like to rank the sentences in the documents to maximize answer recall and precision in the topmost $K << N$, in order to minimize the parsing and subsequent processing. In this stage we set K=300. We assign goodness score to a sentence according to the following criteria:

1. Exact match of proper names such as “Denver” and “Aspen” get 100% bonus score.
2. Upper case term match of length greater than 1 get 60% bonus, otherwise get 30%. For example, match of “United States” is better than just of “United”.
3. Lower case matches get the original score.
4. Lower case term match with WordNet expansion stems get 10% discount. If the original term is capital case then it gets 50% discount. For example, when *Cag(e)* matches *cag(e)*, the former may be the last name of some person while the latter is an object; therefore, the case mismatch signals less reliable information.
5. Lower case term matches after Porter stemming get 30% discount. If the original term is capital case then 70% discount. The Porter stemmed match is considered less reliable than a WordNet stem match.
6. Porter stemmer matches of both question word and sentence word get 60% discount. If the original term is capital case then get 80% discount.

7. If CONTEX indicates a term as being QSUBUMED then it gets 90% discount. For example, “Which country manufactures weapons of mass destruction?” where “country” will be marked as *qsubsumed*.

Normally common words are ignored unless they are part of a phrase in question word order. Based on these scores, the total score for a sentence is:

Sentence score = sum of word scores

At the end of the ranking we apply qtarget filtering to promote promising answer sentences. For example, since the question “How far is it from Denver to Aspen?” is asking for a distance quantity, any sentence that contains only “Denver” or “Aspen” but not any distance quantities are thrown out. Only the top 300 remaining sentences are passed to the answer pinpointing module.

The bonus and discount rates given here are heuristics. We are in the process of developing mechanisms to learn these parameters automatically.

4. Answer Matching using Qtarget-Specific Knowledge

Once the candidate answer passages have been identified, their sentences are parsed by CONTEX. The Matcher module then compares their parse trees to the parse tree of the original question. The Matcher performs two independent matches (Hovy et al., 2001; Hovy et al., 2000):

- match qtargets and qargs/qwords in the parse trees,
- match over the answer text using a word window.

Obviously, qtargets and their accompanying qargs play an important role; they enable the matcher to pinpoint within the answer passage the exact, syntactically delimited, answer segment. (In contrast, word window matching techniques, that have no recourse to parse structures, have no accurate way to delimit the exact answer boundaries.)

Unfortunately, there are many questions, for which the qtarget (which can be as generic as NP), syntactic clues and word overlap are insufficient to select a good answer. Over the past year we therefore focused on strategies for dealing with this, and developed the following.

Expected Answer Range

For quantity-targeting questions, humans often have a good sense of reasonable answer ranges and would find it easy to identify the correct answer in the following scenario:

Q: What is the population of New York?

S1. The mayor is held in high regards by the 8 million New Yorkers.

S2. The mayor is held in high regards by the two New Yorkers.

Even without any knowledge about the population of specific cities and countries, a population of 8,000,000 makes more sense than a population of 2. We mirror this ‘common sense’ knowledge by biasing quantity questions like the one above towards normal value ranges.

Abbreviation Knowledge

Multi-word expressions are not abbreviated arbitrarily:

Q: What does NAFTA stand for?

S1. This range of topics also includes the North American Free Trade Agreement, NAFTA, and the world trade agreement GATT.

S2. The interview now changed to the subject of trade and pending economic issues, such as the issue of opening the rice market, NAFTA, and the issue of Russia repaying economic cooperation funds.

After Webclopedia identifies the qtarget of the question as I-EN-ABBREVIATION-EXPANSION, the system extracts possible answer candidates, including “North American Free Trade Agreement” from S1 and “the rice market” from S2. Based on the perfect match of the initial letters of the first candidate with the acronym NAFTA, an acronym evaluator easily prefers the former over the latter candidate.

Semantic Mark-Up in Parse Trees

Phone numbers, zip codes, email addresses, URLs, and different types of quantities follow patterns that can be exploited to mark them up, even without any explicit mentioning of key words like "phone number". For a question/sentence candidate pair like

Q: What is the zip code for Fremont, CA?

S1. From Everex Systems Inc., 48431 Milmont Drive, Fremont, CA 94538.

Webclopedia identifies the qtarget as C-ZIP-CODE. To match such qtargets, the CONTEX parser marks up (likely) zip codes, based on both structure (e.g., 5 digits) and context (e.g., preceding state code). Two more question/answer pairs that are matched this way:

Q: What's Dianne Feinstein's email address?

Qtarget: C-EMAIL-ADDRESS

S1. Comments on this issue should be directed to Dianne Feinstein at senator@feinstein.senate.gov

Q: How hot is the core of the earth?

Qtarget: I-EN-TEMPERATURE-QUANTITY

S1. The temperature of Earth's inner core may be as high as 9,000 degrees Fahrenheit (5,000 degrees Celsius).

Using External Glosses for Definition Questions

We have found a 10% increase in accuracy in answering definition questions by using external glosses.

Q: What is the Milky Way?

Candidate 1: outer regions

Candidate 2: the galaxy that contains the Earth

For the above question, Webclopedia identified two leading answer candidates. Comparing these answer candidates with the gloss that the system finds in Wordnet:

Wordnet: Milky Way—the galaxy containing the solar system

Webclopedia biases the answer to the candidate with the greater overlap, in this case clearly "the galaxy that contains the Earth".

Finding Support For a Known Answer

It seems against all intuition that a question like

Q1: What is the capital of the United States?

initially poses great difficulties for a question answering system. While a question like

Q2: What is the capital of Kosovo?

can easily be answered from text such as

S2. ... said Mr Panic in Pristina, the capital of Kosovo, after talks with Mr Ibrahim Rugova ...

many American readers would find a newspaper sentence such as

S1. Later in the day, the president returned to Washington, the capital of the United States.

almost insulting. The fact that Washington is the capital of the United States is too basic to be made explicit. In this unexpectedly difficult case, we can fall back on sources like Wordnet:

Wordnet: Washington—the capital of the United States

which, as luck would have it, directly answers our question. Based on this knowledge, Webclopedia produces the answer, represented as a lexical target (LEX "Washington"), which the IR module then uses to focus its search on passages containing "Washington", "capital" and "United States". The matcher then limits the search space to "Washington". The purpose of this exercise is not as ridiculous as it might first appear: even though the system already knows the answer before consulting the document collection, it makes a contribution by identifying documents that support "Washington" as the correct answer.

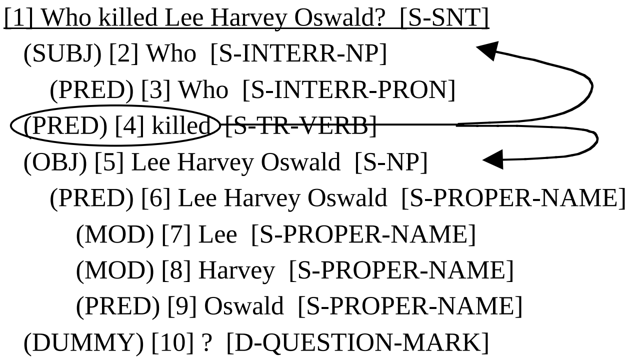
Semantic Relation Matching in Webclopedia

In question answering, matching words and groups of words is often insufficient to accurately score an answer. As the following examples demonstrate, scoring can benefit from the correct matching of semantic relations in addition:

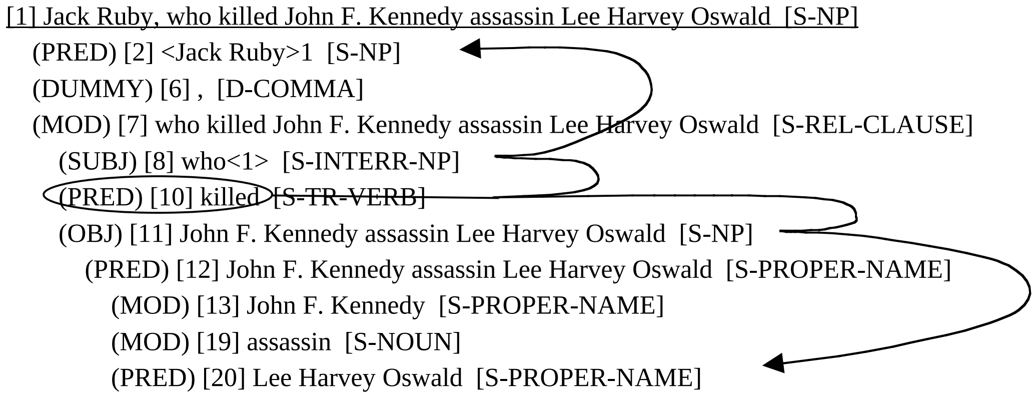
> Question 110: Who killed Lee Harvey Oswald?
> Qtargets: I-EN-PROPER-PERSON&S-PROPER-NAME, I-EN-PROPER-ORGANIZATION (0.5)
>
> S1. Belli's clients have included **Jack Ruby**, who killed John F. Kennedy assassin Lee Harvey Oswald, and Jim and Tammy Bakker. [Score: 666.72577; 07/25/90; LA072590-0163]
>
> S2. On Nov. 22, 1963, the building gained national notoriety when Lee Harvey Oswald allegedly shot and killed **President John F. Kennedy** from a sixth floor window as the presidential motorcade passed.
> [Score: 484.50128; 10/31/88; AP881031-0271]

Note: Answer candidates are bold ("red"), while constituents with corresponding words in the question are underlined ("blue") (http://www.isi.edu/natural-language/projects/webclopedia/sem-rel-examples.html).

Both answer candidates S1 and S2 receive credit for matching "Lee Harvey Oswald" and "kill", as well as for finding an answer (underlined) of the proper type (I-EN-PROPER-PERSON), as determined by the qtarget. However, is the answer "Jack Ruby" or "President John F. Kennedy"? The only way to determine this is to consider the semantic relationship between these candidates and the verb "kill", for which Webclopedia uses the following question and answer parse trees (simplified here):



```
[1] Who killed Lee Harvey Oswald?  [S-SNT]
  (SUBJ) [2] Who  [S-INTERR-NP]
    (PRED) [3] Who  [S-INTERR-PRON]
  (PRED) [4] killed  [S-TR-VERB]
  (OBJ) [5] Lee Harvey Oswald  [S-NP]
    (PRED) [6] Lee Harvey Oswald  [S-PROPER-NAME]
      (MOD) [7] Lee  [S-PROPER-NAME]
      (MOD) [8] Harvey  [S-PROPER-NAME]
      (PRED) [9] Oswald  [S-PROPER-NAME]
  (DUMMY) [10] ?  [D-QUESTION-MARK]
```



```
[1] Jack Ruby, who killed John F. Kennedy assassin Lee Harvey Oswald  [S-NP]
  (PRED) [2] <Jack Ruby>1  [S-NP]
  (DUMMY) [6] ,  [D-COMMA]
  (MOD) [7] who killed John F. Kennedy assassin Lee Harvey Oswald  [S-REL-CLAUSE]
    (SUBJ) [8] who<1>  [S-INTERR-NP]
    (PRED) [10] killed  [S-TR-VERB]
    (OBJ) [11] John F. Kennedy assassin Lee Harvey Oswald  [S-NP]
      (PRED) [12] John F. Kennedy assassin Lee Harvey Oswald  [S-PROPER-NAME]
        (MOD) [13] John F. Kennedy  [S-PROPER-NAME]
        (MOD) [19] assassin  [S-NOUN]
        (PRED) [20] Lee Harvey Oswald  [S-PROPER-NAME]
```

For S1, based on these parse trees, the matcher awards additional credit to node [2] (Jack Ruby) for being the logical subject of the killing (using anaphora resolution) as well as to node [20] (Lee Harvey Oswald) for being the head of the logical object of the killing. Note that—superficially—John F. Kennedy appears to be closer to “killed”, but the parse tree correctly records that node [13] is actually not the object of the killing. The candidate in S2 receives no extra credit for semantic relation matching.

Robustness

It is important to note that the Webclopedia matcher awards extra credit for *each* matching semantic relationship between two constituents, not only when everything matches. This results in robustness that comes in handy in cases such as:

> Question 268: Who killed Caesar?
> Qtargets: I-EN-PROPER-PERSON&S-PROPER-NAME, I-EN-PROPER-ORGANIZATION (0.5)
>
> S1. This version of the plot to kill Julius Caesar is told through the eyes of **Decimus Brutus**, the protege whom Caesar most trusted and who became one of his assassins.
> [Score: 284.945; 93/05/15; FT932-8961]
>
> S2. Having failed to prevent Cleopatra’s henchwoman Ftatateeta from killing **Pothinus**, Caesar lets Rufius—the new governor of Egypt—murder her, before turning his back on the lot of them in a devastating display of political indifference. [Score: 264.30093; 92/02/06; FT921-10331]

In S1, the matcher gives points to Caesar for being the object of the killing, but (at least as of now) still fails to establish the chain of links that would establish Brutus as his assassin. The predicate-object credit however is enough to make the first answer score higher than in S2, which, while having all agents right next to each other at the surface level, receives no extra credit for semantic relation matching.

Good Generalization

Semantic relation matching applies not only to logical subjects and objects, but also to all other roles such as location, time, reason, etc. It also applies at not only the sentential level, but at all levels:

> Question 248: What is the largest snake in the world?
> Qtargets: I-EN-ANIMAL
>
> S1. **Reticulated pythons** are the world’s largest snakes, reaching lengths of up to 36 feet.
> [Score: 384.42365; 12/08/88; AP881208-0148]
>
> S2. The amazing Amazon, the widest, wettest and, so National Geographic now affirms, the longest river in the world (4,007 miles, 51 longer than the Nile), boasts the longest snake **the most venomous viper**, the biggest rat, beetle and ant, along with razor-toothed piranhas that can reduce a Brahman steer to raw bones in minutes and electric eels delivering 640 volts, enough to drive a Metro-North commuter train. [Score: 291.98352; 02/29/88; AP880229-0246]

In the S1, [world] receives credit for modifying snake, even though it is the (semantic) head of a post-modifying prepositional phrase in the question and the head of a pre-modifying determiner phrase in the answer sentence. While the system still of course prefers “in the world” over “the world’s” on the constituent matching level, its proper relationship to snake (and the proper relationship between “largest” and “snakes”, as well as “pythons” and “snakes”) by far outweigh the more literal match of “in the world”.

Using a Little Additional Knowledge

Additionally, Webclopedia uses its knowledge of the semantic relationships between concepts like “to invent”, “invention” and “inventor”, so that in example 209, “Johan Vaaler” gets extra credit for being a likely logical subject of “invention”, while “David” actually loses points for being outside of the clausal scope of the inventing process in the second case.

> Question 209: Who invented the paper clip?
> Qtargets: I-EN-PROPER-PERSON&S-PROPER-NAME, I-EN-PROPER-ORGANIZATION (0.5)
>
> S1. The paper clip, weighing a desk-crushing 1,320 pounds, is a faithful copy of **Norwegian Johan Vaaler**’s 1899 invention, said Per Langaker of the Norwegian School of Management.
> [Score: 381.0031; 10/09/89; AP891009-0048]

S2. “Like the guy who invented the safety pin, or the guy who invented the paper clip,” **David** added. [Score: 236.47534; 07/20/89; LA072089-0033]

Question 3: What does the Peugeot company manufacture?
Qtargets: S-NP, S-NOUN

S1. Peugeot intends to manufacture **10,000 cars** there each year.
[Score: 360.49545; 10/09/89; AP891009-0048]

S2. These include Coca Cola and Pepsico, the US soft drinks giants, Peugeot the French **car** manufacturer, finance companies GE Capital and Morgan Stanley, Nippon Denro, the Japanese steel manufacturer, and the Scotch whisky maker Seagram and United Distillers, the spirits arm of Guinness. [Score: 323.76758; 93/06/25; FT932-902]

In S2, “car” gets credit as a likely logical object of the manufacturing process, and “Peugeot”, being recognized as a “manufacturer”, is boosted for playing the proper logical subject role. This example shows that particularly when the qtarget doesn’t help much in narrowing down the answer candidate space, semantic relation matching can often make the crucial difference in finding the right answer.

5. Experiments and Results

We entered the TREC-10 QA track, and received an overall Mean Reciprocal Rank (MRR) score of 0.435, which puts Webclopedia among the top performers. The average MRR score for the main task is about 0.234. The answer rank distribution is shown in Figure 2. It indicates that we cannot find answers in the top 5 in about 43% of the cases. Once we find answers we usually rank them at the first place.

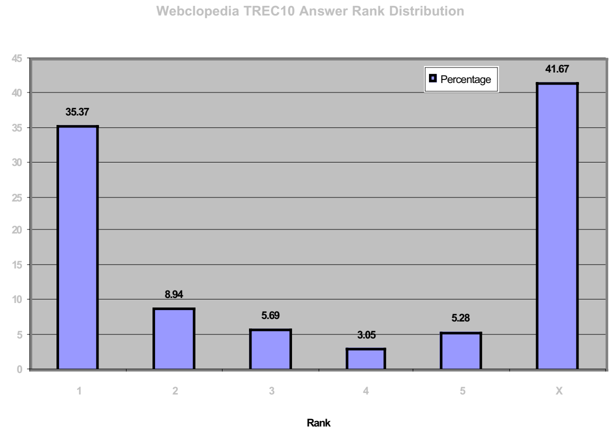


Figure 2. Webclopedia answer rank distribution in TREC-10.

Analysis of the answers returned by the TREC assessors revealed several problems, ranging from outright errors to judgments open to interpretation. One example of an error is a ruling that the answer to “what is cryogenics?” is not “engineering at low temperature” (as defined for example in Webster’s Ninth New Collegiate Dictionary, and as appears in the TREC collection), but rather the more colloquial “freezing human being for later resustication” (which also appears in the collection). Although Webclopedia returned both, the correct answer (which it preferred) was marked wrong. While we recognize that it imposes a great administrative burden on the TREC QA administrators and assessors to re-evaluate such judgments, it is also clearly not good R&D methodology to train systems to produce answers that are incorrect but colloquially accepted. (Checking whether their knowledge is correct is precisely one of the reasons people need QA systems!) We therefore propose an appeals procedure by which the appellant must provide to the administrator the question, the correct answer, and proof, drawn from a standard reference work, of correctness. The administrator can provide a list of acceptable reference works beforehand, which should include dictionaries, lists of common knowledge facts (the seven wonder of the world, historical events, etc.), abbreviation lists, etc., but which would presumably not include local telephone books, etc. (thereby ruling out local restaurants as answer to “what is the Taj Mahal?”).

6. References

Bikel, D., R. Schwartz, and R. Weischedel. 1999. An Algorithm that Learns What’s in a Name. *Machine Learning—Special Issue on NL Learning*, 34, 1–3.

Fellbaum, Ch. (ed). 1998. *WordNet: An Electronic Lexical Database*. Cambridge: MIT Press.

Hermjakob, U. 1997. *Learning Parse and Translation Decisions from Examples with Rich Context*. Ph.D. dissertation, University of Texas at Austin. file://ftp.cs.utexas.edu/pub/ mooney/papers/hermjakob-dissertation-97.ps.gz.

Hermjakob, U. 2000. Rapid Parser Development: A Machine Learning Approach for Korean. In *Proceedings of the North American chapter of the Association for Computational Linguistics* (NAACL-2000). http://www.isi.edu/~ulf/papers/kor_naacl00.ps.gz.

Hermjakob, U. 2001. Parsing and Question Classification for Question Answering. In *Proceedings of the Workshop on Question Answering at the Conference ACL-2001*. Toulouse, France.

Hovy, E.H., L. Gerber, U. Hermjakob, M. Junk, and C.-Y. Lin. 2000. Question Answering in Webclopedia. *Proceedings of the TREC-9 Conference*. NIST. Gaithersburg, MD.

Hovy, E.H., U. Hermjakob, C.-Y. Lin, and D. Ravichandran. 2001. Toward Semantics-Based Answer Pinpointing. *Proceedings of the Human Language Technologies Conference* (HLT). San Diego, CA.

Witten, I.H., A. Moffat, and T.C. Bell. 1994. Managing Gigabytes: Compressing and indexing documents and images. New York: Van Nostrand Reinhold.