INTERVENTO

MALVINA NISSIM[○] – JOHAN BOS[◇]
([○]ISTC-CNR, Roma – [◇]Università La Sapienza, Roma)

Using the Web as a Corpus in Natural Language Processing

1. INTRODUCTION

Research in Natural Language Processing (*NLP*) has in recent years benefited from the enormous amount of raw textual data available on the World Wide Web. The presence of standard search engines has made this data accessible to computational linguists as a *corpus* of a size that had never existed before. Although the amount of readily available data sounds attractive, the Web as a *corpus* has the inherent disadvantage of being unstructured, uncontrollable, unprocessed, constantly changing, and of heterogeneous quality. These conflicting aspects force *NLP* researchers to develop new and creative techniques to exploit it as a linguistic resource. In this paper we illustrate how the Web can support *NLP* tasks, focusing on two applications: acquiring semantic knowledge, and validating linguistic hypotheses.

As the largest *corpus* available, the Web can be used as a source for extracting lexical knowledge by means of specifically task-tailored syntactic patterns. One general example is the extraction of hyponymic relations by means of ISA-like patterns (see Hearst, 1992). Example applications that benefit from this are named entity recognition, ontology building/editing, and anaphora resolution. The Web can also be used as a testbed for linguistic hypotheses and/or evaluation of system outputs. For example, the frequency of different PP-attachments produced by a parser can be compared, or the frequency of different spelling variants.

To access and retrieve information from the Web, *NLP* researchers often use off-the-shelf interfaces provided by major search engines such as *Google*, *Yahoo*, and *AltaVista*. We will illustrate the general approach by giving two case studies based on our own recent work: one describing how to use the Web for finding antecedents of anaphoric expressions (Section 2), and one that uses the Web in an answer re-ranking task within a question answering system (Section 3). In both of these studies we used the *Google API* to retrieve information from the Web. Finally, we will discuss outstanding issues with using the Web as a *corpus* and possible directions for future work.

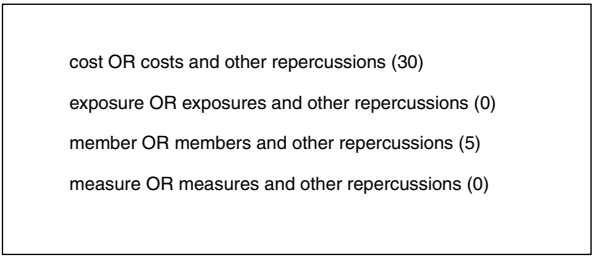
2. Candidate Antecedent Selection for Anaphora Resolution

Other-anaphora is an example of lexical anaphora, where the modifiers *other* or *another* provide a set-complement to an entity already evoked in the discourse model. In Example (1), the NP *other, far-reaching repercussions* refers to a set of repercussions excluding increasing costs, and can be paraphrased as *other (far-reaching) repercussions than (increasing) costs*.

> *(1) In addition to increasing costs as a result of greater financial exposure for members, these measures could have* ***other, far-reaching repercussions****. (Wall Street Journal)*

For interpreting other, far-reaching repercussions as *repercussions other than costs* a resolution system needs the knowledge that costs are or can be seen as repercussions. Most systems that handle this and similar anaphoric phenomena rely on handcrafted resources of lexico-semantic knowledge, such as the *WordNet* lexical hierarchy (Fellbaum, 1998). The alternative method that has been suggested by Markert and Nissim (2005) is to use the Web, to be mined with specifically tailored lexico-syntactic patterns. More specifically, the implicit hyponymy relation between an anaphor (Y) and its antecedent (X) is made explicit via a specific pattern, such as *Y(s) and other Xs*, which indeed indicates that Y is a hyponym of X. For each anaphor, all base noun phrases occurring in an N-sentence window (given the anaphor) are tested in the pattern, and all resulting phrases are submitted as quoted queries to *Google*. (In the example queries below "OR" is the boolean operator and has scope on the previous term only.) As the final resolution step, the most frequent phrase is chosen as the one containing the correct antecedent.

In Example 1, the available noun phrases (considering just a one-sentence window for the sake of clarity) are *increasing costs*, *greater financial exposure*, *members*, *these measures*. Discarding modification and determination, the following phrases are created and searched on *Google*:



The figures in brackets correspond to the number of hits returned by *Google* for each phrase. In this case, *costs* is correctly selected as the antecedent for *repercussions*. Markert and Nissim (2005) show that on this task, as well as on a full NP coreference resolution task, this method works even better than using hyponymy relations in *WordNet*. The same method based on mining *corpora* by means of lexico-syntactic patterns was also tried on the *BNC*, a much smaller (one hundred million words) but controlled and virtually noise-free *corpus* of *British English*. Due to the insufficient size of the *corpus*, results were worse than those obtained using the same technique on the Web, and those obtained using *WordNet* as a source of knowledge.

Although successful on many examples, the Web-based method for *other*-anaphora resolution we have described has some intrinsic limitations. Consider Examples 2-4:

(2) *The move is designed to more accurately reflect the value of products and to put steel on a more equal footing with* ***other commodities***.

(3) *J.L. Henry & Co., Miami, and a principal of the firm, Henry I. Otero of Miami, were jointly fined $30,000 and expelled, for alleged improper use of a customer's funds, among* ***other things***.

(4) *Coleco bounced back with the introduction of the Cabbage Patch dolls. [...] But as the craze died, Coleco failed to come up with* ***another winner.***

All of the three anaphors in the examples above, namely *other commodities*, *another winner*, and *other things*, were wrongly resolved by the Web-based algorithm. Each of these cases highlights a limitation of the approach. For instance, in (2), the distractor *products*, when combined with *commodities* in the standard pattern, yielded a larger number of hits than the correct antecedent *steel*. Various experiments with normalisation methods that would take into account the higher frequency of *products* on its own have not shown good results (Markert and Nissim, 2005). One possible direction for improvement would be the integration of syntactic information and sentence structure. The noun phrases *other things* in (3) shows how sometimes semantically empty heads (such as *thing*) can be used as anaphors; a method that relies entirely on semantic information for anaphora resolution is bound to fail in such cases, since it cannot exploit any mutual (lexico-semantic) constraints holding between anaphor and antecedent. Finally, (4) is a beautiful example of a highly context-dependent relation between the anaphor (*winner*) and the antecedent (*dolls*). Although it is a perfectly acceptable relation in the example above, dolls are not normally defined as winners, and finding evidence of this in a *corpus* – even as large as the Web – is a tricky matter.

3. Answer Re-ranking in "Question Answering"

One of the problems in automatic Question Answering is to choose the most likely candidate "out" of a set of possible answers produced by a *QA system*. The open domain *QA system QED* in (Ahn *et al*., 2005) outputs a set of plausible answers, ranked using a simple scoring method. However, in many cases this ranking can be considerably improved. An example is the question from the *TREC-2005* evaluation campaign *Where was Guthrie born?* with respect to the topic *Woody Guthrie*, for which *QED* initially produced the following ranked answer candidates from the Acquaint newswire *corpus*:

1. Britain
2. Okemah, Okla.
3. Newport
4. Oklahoma
5. New York

Although in this N-best list answers 2 and 4 are correct, in the *TREC* evaluation systems must return one answer only, and as a consequence, this would lead to an incorrect response (Britain) to the question. To deal with this problem, we experimented with additional external knowledge obtained from the Web in order to generate a possibly more accurate ranking. This technique is also known as 'answer validation' or 'sanity checking', and can be seen as a tie-breaker between the top-N answers. In more detail, this method works as follows. For each of the N-best answer candidates, we take the semantic representation of the question (see Ahn *et al*., 2005), and for each answer candidate we generate a set of declarative sentences (covering all morphological variations). The generated sentences are submitted as strict (within quotes) queries for *Google*. Any information from the given question topic which is not included in the generated sentence (for this example *Woody*) is added as a query term in order to constrain the search space. The queries and number of hits returned for each of the queries (in brackets) are shown below.

1. *Woody "Guthrie born in Britain" (0)*
   Woody "Guthrie are OR is OR was OR were born in Britain" (0)
2. *Woody "Guthrie born in Okemah, Okla." (1)*
   Woody "Guthrie are OR is OR was OR were born in Okemah, Okla." (10)

3. *Woody "Guthrie born in Newport" (0)*
   Woody "Guthrie are OR is OR was OR were born in Newport" (0)
4. *Woody "Guthrie born in Oklahoma" (7)*
   Woody "Guthrie are OR is OR was OR were born in Oklahoma" (42)
5. *Woody "Guthrie born in New York" (0)*
   Woody "Guthrie are OR is OR was OR were born in New York" (2)

The returned *Google*-counts are used as the deciding factors to re-rank the N-best answers. Note that we generate several queries for each answer candidate and we sum the returned hits. In this example, the answers would be re-ranked as follows:

1. Oklahoma (7 + 42)
2. Okemah, Okla. (1 + 10)
3. New York (0 + 2)
4. Britain (0 + 0)
5. Newport (0 + 0)

For this example, re-ranking correctly promoted *Oklahoma* to best answer. The success of this approach depends on the performance of the aforementioned generation component, which produces declarative forms of the question. An imperfect generation component might produce non-grammatical or extremely general or vague English sentences. As a consequence there might be no or wrong matches on web pages satisfying the queries. We would like to illustrate this problem by providing another example. Consider the *TREC 2005* question *When was the company founded?* with respect to the topic *Harley-Davidson*. Here the initial answer candidates of *QED* were:

1. 1957
2. 1997
3. 1979
4. 1999
5. 1903

For this question, answer 5 is correct. The generated queries and the number of returned hits are as follows:

1. Harley-Davidson “The company is OR are OR was OR were founded in 1957 (0)”
2. Harley-Davidson “The company is OR are OR was OR were founded in 1997 (48)”
3. Harley-Davidson “The company is OR are OR was OR were founded in 1979 (20)”
4. Harley-Davidson “The company is OR are OR was OR were founded in 1999 (28)”
5. Harley-Davidson “The company is OR are OR was OR were founded in 1903 (38)”

As is clear from these *Google* results, the correct answer (1903) is ranked at number 2 and an incorrect answer (1997) is re-ranked to best answer. This is due to the fact that the generated declarative sentence is far too general. Adding the topic phrase *Harley-Davidson* to constrain the search space is not sufficient to yield the correct answer.

Only resolving the definite description *the company* and subsequently generating as query *Harley-Davidson is OR are OR was OR were founded in* * might solve this problem.

4. Discussion

Although the two examples of applications illustrate successful usages of the Web for *NLP* tasks, there are several obstacles. There are only a limited number of search engines available (*Google*, *Yahoo*, and *Altavista*), so the *NLP* researchers who want to exploit the web as a corpus are dependent on them. Some search engines offer more expressive power in queries (‘near’-operator, *AltaVista*), others a larger range of documents (*Google*). Searching on exact strings and punctuation is restricted, and also the number of queries is limited to a couple of thousand a day. Further, it will be important to distinguish between useful webpages and those that are not trustworthy (from a linguistic as well as content point of view) – of course not everything that is written is correct. Finally, it is virtually impossible to reproduce exact experiments, since the Web is constantly changing. In order to (partially) overcome the noise-related problems as well as those connected with the non-reproducibility of experiments at different times, some researchers have suggested to *dump* a (balanced) large collection of web-pages at a given time and process it with standard *NLP* tools (see for example Baroni and Kilgarriff, 2006).

Overall, by means of the discussion on two different phenomena that we presented, we aimed to illustrate that although a shallow web-based approach to answer-re-ranking might seem attractive, does work to a certain extent and can be used in an nearly effort-free manner (no processing is required) to achieve reasonable results, systems that want to boost their performance in terms of precision must complement this method with further linguistic knowledge.

REFERENCES

Ahn K., Bos J., Curran J. R., Kor D., Nissim M., Webber B., 2005, Question Answering with QED at TREC-2005, in E. Voorhees, L. P. Buckland (eds.), *The Fourteenth Text Retrieval Conference*, Gaithersburg MD, TREC 2005.

Baroni M., Kilgarriff A., 2006, Large Linguistically-Processed Web Corpora for Multiple Languages, in *Conference Companion of EACL 2006 (11th Conference of the European Chapter of the Association for Computational Linguistics)*, *East Stroudsburg PA*, ACL: 87-90.

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

Hearst M., 1992, Automatic Acquisition of Hyponyms from Large Text Corpora, in *Proceedings of the 15th International Conference on Computational Linguistics, Nantes, France*: 539-545.

Markert K., Nissim M., 2005, Comparing Knowledge Sources for Nominal Anaphora Resolution, in "Computational Linguistics", 31(3): 367-402.