Automatic Acquisition of Hyponyms and Meronyms from Question Corpora

Håkan Sundblad[1]

Abstract. We explore how lexical and ontological relations can be acquired automatically from natural language questions. The focus in this paper is on identifying hyponym and meronym relations by using simple pattern matching. It is shown that natural language questions can provide a significant source for ontological information.

1 INTRODUCTION

Within the lexical and ontology engineering communities, it has been recognized that natural language texts provide a rich source for extracting semantic relations, such as hyponyms and meronyms. For instance, Hearst [1] has studied how hyponym relations can be acquired automatically using linguistic patterns such as those listed below[2]:

1. such NP as NP ,* (or | and) NP
2. NP , NP* , or other NP
3. NP , including NP , or | and NP

By running an acquisition algorithm that utilized these patterns on different texts, a number of new hyponym relations were found that were not incorporated into WordNet [4]. The drawback of using large free text corpora is that it is difficult to find patterns with high precision. One way to overcome this is to look at a more restricted type of language. The research in this paper explores one such kind of corpora, namely questions formulated in natural language. Such corpora are interesting from a number of different perspectives. First, natural language questions tend to be more concise and structured than natural language in general. Second, they reflect users' needs and interests, and therefore also reflects what information that should be represented in ontologies and lexical resources. Finally, as question answering systems are becoming publicly available, such corpora will be readily available and large, and therefore become an important source of information.

If we now turn to natural language questions and start by intuition, a question like "Where is Belize located?" contains the information that "Belize" is a location. Similarly, from the question "How many hexagons are on a soccer ball?" we could infer that footballs have hexagons on them, if we do not already have this information. The former question contains a hyponym, while the latter contains a meronym. In this paper we present results from utilizing such patterns for acquiring ontological relations.

[1] Department of Computer and Information Science, Linköping university, Linköping, SE-581 83, Sweden, email: hakjo@ida.liu.se

[2] The syntax is adopted from Hearst [1]

2 METHOD

This section describes how the experiments were conducted, and what material that was used.

2.1 The Corpus

The corpus utilized in these investigations is the test corpus used in the question answering track of TREC9[3] (Text REtrieval Conference) and contains 692 questions. The corpus consists of factual questions that have answers and are not restricted to any specific domain.

2.2 Analysis

The corpus was first analyzed manually, and potentially interesting patterns were noted. The productivity of these patterns were then explored using common UNIX tools that incorporates regular expressions [3], such as egrep and sed. Perl has also been used for some, more complex, operations.

Initially, every question was annotated as to whether they contained any hyponym or meronym relations. Of the 692 questions, 112 were judged to contain a hyponym relation, and 125 contained a meronym relation. No cross-evaluation was performed on these judgements.

3 ACQUIRING HYPOMYMS

This section explores how hyponym relations can be extracted from the corpus of questions. We will start by examining the most simple and obvious cases, and then proceed to investigate gradually more complex ones.

Perhaps the most obvious hyponyms that can be extracted from the corpus is hyponyms to the ontological category <person>. These are for instance realized in a pattern like:

H1 Who is/was X[4]

These patterns differ from the kind that Hearst [1] uses in that they are directed at finding hyponyms to the category <person>; they are not a general pattern for finding hyponyms. A more complex, but still productive, variation is different realizations of the following pattern:

H2 What is/was X best/most known/famous for?

[3] The corpus can be found on `http://trec.nist.gov/data.html`

[4] The syntax used in this paper is a sloppy kind of regular expressions in order not to obscure the patterns proper. X and Y translates to the regular expression /.+/, i.e., any character repeated one or more times.

In theory, at least, this pattern is too general, i.e., not all X:s will be persons.

Locations are another category to which hyponyms can be extracted with quite simple patterns. The prototypical patterns being:

H3 Where is X (located)?

H4 What is the location of X?

A common type of question concerns what different acronyms or abbreviations stand for. Not surprisingly, these questions are often formulated as:

H5 What does X stand for?

H6 X is an abbreviation/acronym for what?

From this we can infer that X is an <acronym>. This of course presupposes that there is a category in the ontology labeled acronym which represents both acronyms and abbreviations. This, of course, might not be the case for most ontologies, but it is the gist of the approach that this paper intends to capture. Extracting acronyms can be particularly useful as new ones appear almost on a daily basis.

The only pattern found in the corpus that exhibits the generality of those in Hearst [1] is:

H7 What kind/type of Y is/was X?

This pattern identifies hyponyms such as:

- hyponym("Winnie the Pooh", "animal")
- hyponym("the Wisconsin Badgers", "a sports team")
- hyponym("the Buffalo Sabres", "sports team")
- hyponym("the Golden Gate Bridge", "bridge")

The quality of these hyponyms suggests that this is can be a particularly useful pattern for finding general hyponyms.

3.1 Results

The productivity of the patterns described above are presented in table 1. The first column contains the pattern ID, the second column describes the pattern, and the third column presents the number of hyponyms found (found/correct/unique/erroneous).

Id	Pattern	Productivity
H1	Who is/was X?	20 / 20 / 20 / 0
H2	What is/was X best/most known/famous for?	9 / 8 / 8 / 1
H3	Where is X (located)?	40 / 38 / 36 / 2
H4	What is the location of X?	2 / 2 / 2 / 0
H5	What does X stand for?	8 / 8 / 8 / 0
H6	X is an/the acronym/abbreviation for what?	3 / 3 / 2 / 0
H7	What kind/type of Y is/was X?	4 / 4 / 3 / 0

Table 1. Results of hyponym extraction.

As can be seen in table 1, pattern H1 is the most productive, identifying 20 unique names. Simple hyponyms like hyponym("Peter Weir", "person") might not be very exciting, but for ontology engineering this can useful. One of the biggest challenges for ontologies is to keep up with the real world. Thus, automatically acquiring new names with a high precision is of great importance.

Pattern H2 generates one false positive, which is due to the question "What is Black Hills, South Dakota most famous for?". Using a larger corpora, the number could be significantly higher. It is difficult to modify the pattern to discriminate between persons and locations, as the realizations can be identical on the surface. One solutions would be to claim that the hypernym category that the pattern identifies should be sought at a higher level.

If we turn to patterns identifying locations, pattern H3 generates only two false positives out of 38 correct. Both of the questions giving rise to these regards the location of the corpus callosum, which in some sense is a location, just not in the sense that ontologies usually considers them.

The only pattern for finding general hyponyms, pattern H7, finds four hyponyms, of which three are unique kinds of hyponyms. This might admittedly not seem very impressive. But if we consider that we are dealing with a corpus consisting of 5,000 words and that Hearst [1] found a total of 330 hyponyms in a 8,600,000 word corpus, we might want to reconsider. Of course, the sample is to small to draw any conclusions, but the results hint that it might be worth investigating a larger corpus.

Using the seven patterns described here captures a total of 83 of the 112 hyponyms judged as extractable in the corpus. Furthermore, 79 of these were unique. This gives a precision of 96.5% and a recall of 74.1%.

4 ACQUIRING MERONYMS

Before we go into detail about patterns for acquiring meronyms, a few words must be said of the term meronym. Miller [4] loosely defines a meronym as a part-whole (HAS-A) relation, and states that a concept x is a meronym of a concept y "if native speakers of English accept sentences constructed from such frames as A y has an x (as part) or An x is a part of y" [4, p. 8]. The prototypical examples usually concern cars that have doors, seats, mirrors. However, in WordNet one can also find that a person has a personality. This meronym differs from others such as legs and arms in that it is a rather abstract notion. Incidentally, it is mostly these kind of abstract parts that constitute a concept that is found using the techniques described here. I will therefore adopt a looser interpretation of meronyms than what might normally be the case. A meronym is therefore considered to something that is a part of a concept, rather than an object.

Extracting meronyms from natural language questions is not as straightforward as hyponyms. It is hard to find general patterns that cover a large set of instances. However, as this section will show, there are a number of useful and productive patterns.

Perhaps the most useful patterns for finding meronyms is:

M1 What is/was the X of Y?

We also include the "What's the X of Y" variant in this pattern. Examples of found meronyms using this pattern are listed below:

- meronym("capital", "Burkina Faso")
- meronym("life expectancy", "an elephant")
- meronym("occupation", "Nicholas Cage")
- meronym("population", "Mozambique")
- meronym("primary language" "the Philippines")
- meronym("salary", "a U.S. Representative")
- meronym("term", "a group of geese")
- meronym("wingspan", "a condor")

Once again, have in mind that the term meronym is used loosely. The second and third most productive patterns for acquiring meronyms are respectively:

M2 What is the X for Y?

M3 What is X's Y?

Another useful pattern for finding meronyms is:

M4 How many X are in/on Y?

There are other variants of questions beginning with "How" that could prove interesting for acquiring meronyms, but given the small corpus the number of hits is usually quite low.

4.1 Results

The productivity of the patterns for acquiring meronyms are presented in table 2. The first column contains the pattern ID, and the second column contains the pattern itself. The third column contains the following information: the total number of hits, the number of unique meronyms (including "name"), and the number of false positives.

Id	Pattern	Productivity
M1	What is/was the X of Y?	89 / 73 / 43 / 16
M2	What is the X for Y?	19 / 17 / 15 / 2
M3	What is X's Y?	14 / 11 / 9 / 3
M4	How many X are in/on Y?	3 / 3 / 3 / 0

Table 2. Results of meronym extraction.

Pattern M1 generated 89 hits in the corpus, i.e., almost 13% of the questions are of this kind. However, around 40% of these basically identify the relation meronym("name", X), i.e., that a part of a concept is that it has a name, which is not too exciting as this holds for all concepts. As for the rest, 84% constitute more or less interesting meronyms. What is common for almost all of the identified meronyms is that they are at a too specific level, i.e., life expectancy is really a part of the concept <living thing>, occupation a part of the concept <person>, and wingspan a part of the concept <bird>.

Pattern M2 exhibits the same problems as M1 and acquires very similar information.

The errors generated using M3 can all be eliminated by forbidding patterns starting with "What is *the*...?".

The pattern M4 does unfortunately not generate nearly as many hits as the former mentioned pattern, but the precision is 100% in the corpus. One of the meronyms it finds is meronym("hexagons", "soccer ball"), which nicely reflects what a user finds interesting about a soccer ball.

Given that the corpus was judged to contain a total of 125 meronyms, and these four patterns covers 104 of those and generates 21 erroneous, we have both a precision and recall of 83.2%.

5 CONCLUSIONS

Using simple patterns like those presented in this paper can yield important and useful information for lexicon and ontology engineering. The benefits of the approach is its simplicity, but this is (perhaps) also the main drawback. Using simple patterns has been proved to be quite powerful and successful for natural language in general, and the benefit of using questions is that these are often more concise, information rich, and structured to greater extent than free texts.

Extracting ontological information from questions is particularly useful for finding persons and locations. This is due to the simple fact that these categories correspond well to the question types *who* and *where*. This allows for patterns that are rather specific and directed toward identifying very specific hyponym relations. This is probably more difficult when one is using large general text corpora [1]. The only pattern for finding hyponyms that in its generality resembles those presented by Hearst [1], was the "What type/kind of X is Y?". In the current corpus only four instances were found, but this might still prove to be a very productive pattern for larger corpora.

The results presented in this paper might have turned out different if another larger and more general question corpus would have been used. As mentioned, the corpus originates from TREC9. Hirschman and Gaizauskas [2] points out that these were limited to simple factual questions that had answers in the document collections used. This has been changed for the TREC10, where questions with no answer and questions with list-answers (e.g., "list the countries bordering to Afghanistan") are also incorporated. Using such a corpus necessarily means that new patterns must be constructed.

6 SUMMARY

This paper has illustrated ways in which natural language questions can be used for ontology engineering. Analyzing questions can prove useful especially for updating an existing ontology with new information, such as new companies, acronyms, or persons. However, the approach can also be used to add meronymic information about objects in the ontology. By looking at real questions formulated by real users, the ontology can better reflect the man-on-the-street's view of the world, rather than some scientific abstraction of it.

7 FUTURE RESEARCH

The most obvious next step is to investigate a significantly larger, and less biased, corpus in order to establish the most common and useful patterns.

A necessary extension to the work presented here is to do more linguistic analyses of the parts of the semantic relations acquired. At the moment, when a pattern matches something, a simple cut and paste operation is performed. However, in ontologies and lexicons we do not want for instance "an elephant" to be a node label, but simply "elephant". This would require some form of stripping of determiners, prenominal modifiers, and other irrelevant information. One way to to accomplish this is to parse the questions, and then extract the heads of the noun phrases.

One possible extension to this approach is looking at how complete question answering systems can be used for ontology engineering. For instance, given that we have a system that answers a question as follows:

```
Q: Who was the first Russian astronaut
   to walk in space?
A: The broad-shouldered but paunchy
   Leonov, who in 1965 became the first
   man to walk in space, signed
   autographs.
```

We would, perhaps, like to infer that Leonov has the property

of being the first Russian astronaut to walk in space. This requires a linguistic component that can identify "Leonov" as being the entity in the answer that has the requested property. One such linguistic tool that we are intending investigate further is the Functional Dependency Grammar (FDG) parser developed by Tapanainen and Järvinen [5]. In the above example, FDG correctly identifies "Leonov" as beeing the subject of the sentence, and hence it would be possible to link the answer to the question.

ACKNOWLEDGEMENTS

This research was supported by the Swedish national Graduate School for Language Technology (GSLT) and the Swedish Agency for Innovation Systems (VINNOVA).

REFERENCES

[1] M. A. Hearst, 'Automatic acquistition of hyponyms from large text corpora', in *Proceedings of the Fourteenth International Conference on Computational Linguistics*, Nantes, France, (1992).

[2] L. Hirschman and R. Gaizauskas, 'Natural language question answering: the view from here', *Natural Language Engineering*, **7**(4), 275–300, (2001).

[3] L. Karttunen, J-P. Chanod, G. Grefenstette, and A. Schille, 'Regular expressions for language engineering', *Natural Language Engineering*, **2**(4), 305–328, (1996).

[4] G. A. Miller, R. Beckwith, C. Fellbaum, D. Gross, and K. Miller. Introduction to wordnet: an on-line lexical database. `http://www.cogsci.princeton.edu/~wn/papers/`, 1993.

[5] P. Tapanainen and T. Järvinen, 'A non-projective dependency parser', in *Proceedings of the 5th Conference on Applied Natural Language Processing (ANLP'97)*, pp. 64–71, Washington, D.C., (1997).