"Automatic Acquisition of Hyponyms from Large Text Corpora," by Marti Hearst, describes a method for using lexico-syntactic patterns on unrestricted, domain-independent text to acquire new hyponyms of nouns. The paper also sketches a procedure to automatically discover new patterns. Finally, it suggests some possible applications, including using the newly-acquired hyponyms to expand WordNet, a lexico-semantic ontology.

One noun, X, is a hyponym of another noun, Y, if X can be said to be a (kind of) Y. Y is called a hypernym of X and X is called a hyponym of Y. For example, both "treasury" and "temple" are hyponyms of "civic building," which is itself a hypernym of the former two. The goal of the method is to use text-matching patterns on large corpora to automatically acquire such relations.

For example, take the pattern

$$NP_0 \text{ such as } \{NP_1, NP_2, ..., (\text{and}|\text{or})\} \ NP_n,$$

where the NP_i's are simple noun phrases. Now consider the sentence fragment

> ... and various animals, such as dogs, pigs and chickens ...

Here, *dogs*, *pigs* and *chickens* are all hyponyms of *animals*, which in turn is a hypernym of the former three. The pattern will recognize NP_0 as *animals*, NP_1 as *dogs*, NP_2 as *pigs* and NP_3 as *chickens*. From this, the program will infer that hyponym(*dog*, *animal*), hyponym(*pig*, *animal*), and hyponym(*chicken*, *animal*) all hold. Notice that all of the nouns been converted to their singular forms. In general, it is desirable to store only the non-inflected form of any noun.

Furthermore, words that modify the nouns are usually discarded, with a few exceptions. If the pattern had stated *pretty chickens*, the relationship hyponym(*pretty chicken*, *animal*) would hold, but it would be better to keep only the more general hyponym(*chicken*, *animal*). On the other hand, keeping modifiers is sometimes necessary for hyponymy to hold: hyponym(*broken bone*, *injury*) is correct but hyponym(*bone*, *injury*) is false.

Six different patterns are presented in the paper, but they are all similar in structure to the one shown above. These patterns are used to find word pairs in large corpora that satisfy the hyponymy relation. The patterns are described using grammar rules for a unification-based constituent analyzer, which builds on the output of a part-of-speech tagger.

The author also outlines a method to discover new patterns. The first step is to decide what lexical relationship we want to find new patterns for. In this case, the relationship is hyponymy, and this is the relationship that seems to work best for this method. Second, gather a list of pairs of terms for which the relationship holds, such as (*broken bone*, *injury*), (*cow*, *animal*) and (*country*, *England*). Next, find places in the corpus where these pairs occur near each other and record the environment. Presumably, what this means is to record syntactic and lexical features about the local context of where each pair occurs; however, what "environment" means here was not stated. The next step is to find what these environments have in common and hypothesize that the common environments indicate the relationship of interest. This step is not completely defined and is currently carried out manually. Finally, once the pattern has been discovered, use it to gather more instances of relationship pairs and use them to search the corpus for more patterns. This method has the potential to be automated. Carrying this out manually, the author was able to find three new patterns.

One application of this is detection of semantically related nouns. Hyponyms are semantically related, however, using these patterns on unrestricted text can also yield pairs that may not be related through typical hyponymy. Consider hyponym(*detonating explosive*, *blasting agent*). This is not a canonical IS-A relation, but these two phrases are nevertheless semantically similar. Thus, this method could prove useful for expanded synonym expansion. The previous example of hyponym(*broken bone*, *injury*) is also an atypical hyponym relation and cannot be found in WordNet. However, with this bit of information, a program can determine that a broken bone is an injury without having to look more deeply into "broken bone."

Another application is expanding the WordNet ontology, which includes a hierarchy of hyponymy relationships between synsets (synonym sets). As of this writing, WordNet contains 82,115 noun synsets and 26,000 noun synsets at the time that Hearst published his paper. However, WordNet is still incomplete and adding words is a manual process. Therefore, a possible use the method presented in this paper to suggest new hyponyms for the hierarchy.

As it turns out, out of an 8.5M-word encyclopedia, 7067 sentences contained "such as," and of these 152 relations were found using a restricted form of the pattern shown above. 180 out of 226 unique words in these relations existed in the WordNet hierarchy and many of the missing words have since been added. Problems with the results included metonymy and underspecified relations, probably from missing context.

Overall, this is a cheap method that can be seen as helping to build natural language processing tools which rely on large amounts of semantic knowledge. One advantage is that it does not rely on pre-encoded knowledge but only simple lexico-syntactic patterns. Unfortunately, number of relations found were low compared to the size of the corpus. The pattern-acquisition algorithm outlined could increase the results, but number of results by using only patterns will probably not be huge.