Applied NLP: Advanced online Information Systems:
Automatic Construction of Linguistic Resources

Sebastian Blohm, 07.06.02

Abstract

This session gives three experiments in the field of processing of linguistic information to acquire knowledge. Two of them building up a resource of linguistic information and one processing natural language data to gain structured information.

Outline

Introduction: Linguistic Resources in Human-Computer Interaction
Experiment 1: Word Categories (Riloff, Shepherd)
Experiment 2: Refining Taxonomies (Ide, Veronis)
Experiment 3: Extraction and Structuring of Information (Embley, Capbell, Smith)
Document search with the help of semantic information
Conclusion/Discussion

Introduction

Means of human-computer-interaction are: GUI, Key-Combinations, programming languages and (seldom) natural language. They can be compared by their expressiveness. NL is extremely powerful but not designed for machine with discrete states and a restricted cognitive architecture: to many degrees of freedom in interpretation, to many ways of references. The user is likely to overestimate the uniqueness of his references.
To get started one needs semantic knowledge:

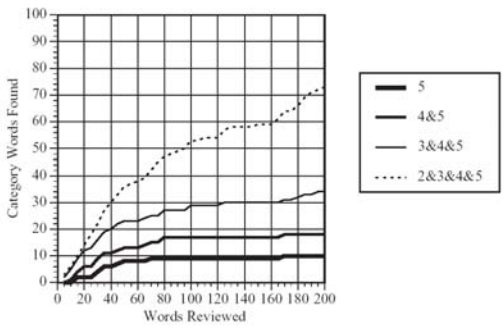
- rules of mapping of NL representation to facts.
- rules of inferences between NL representations.

The latter is much easier but will never produce understanding, thus NLP presently is restricted to heuristic approaches.

Experiments

A Corpus-Base Approach for Building Semantic Lexicons. Riloff, E. and Shepherd, J (1997). Saltlake City. Semi-Automatic corpus-based finding words matching a category indicated through a few "seed words". A list of candidates is generated by a "bootstrapping" algorithm. Non-fitting word are sorted out by hand to optimize recall. Bootstrapping Algorithm:

```
wähle eine Kategorie C
init seedwords
categorywords={}
wiederhole ca. 8 Mal
      finde alle Kontexte der
      Seedwords
      für alle Wörter W, die in
      den Kontexten auftreten:
            Score(W,C) = freq(W im
            Kontext von C) /
            freq(W)
            categorywords =
            categorywords +
            (W,Score(W,C))
      kopiere die 5 besten W in
      seedwords
```

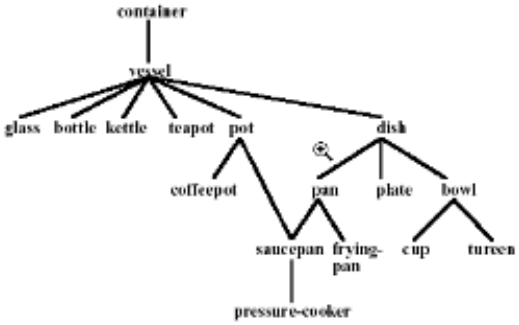


Refining Taxonomies Extracted from Machine Readable Dictionaries. Ide, N. and Véronis, J. (1994). New York. Taxonomies of the hypernym (Oberbegriff) relation are generated using the following heuristic: The hypernym is head of the defining noun phrase or alternatively head of a phrase starting with 'of'.
dipper: a ladle used for dipping
slice: any of various utensils…
In that the following problems cause an error-rate of 55-70%:

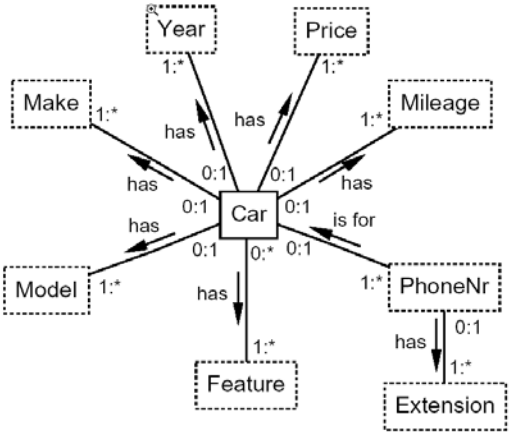
- Attachment too high – description uses a hypernym of the actual hypernym.
- Missing hypernyms – other way of description used.
- Missing overlaps: – out of several possible hypernyms only one is given.
- Or-conjoined heads: – description yields several heads not all being a universal hypernym
- Circularity – words are described by each other.

The problems are overcome (down to 6%) by combining five dictionaries using a simple algorithm:
Take all hypernym candidates for a word and erase those which,

- are or-conjoined.
- are hypernym of another candidate for the same word.



Ontology-Based Extraction and Structuring of Information from Data-Rich Unstructured Documents. Ebley, D., Campbell, D., Smith, R. (1998)
Information from documents of a specific domain that can be described by a small ontology can be extracted from unstructured texts and stored in a database.



Key elements of the example-system's architecture:

Ontology Parser:
Input: Ontology-File.
Output: Constant/Keyword-Rules (Regular Expressions for recognition of constants and keywords), Object/Relationship-Constraints: which constants can be described by which keywords/type of data.

Recognizer
Input: Text, Constant/Keyword-Rules
Output: name/string/position table. All constants and keywords recognized by any RE.

Structured Text Generator :
Input: name-string position-table, object/relationship constraints, heuristics
Output: attribute/value pairs
Assigns attribute-value pairs according to the heuristics. Main elements of the heuristics are keyword proximity and the constraints in the number of possible values for an attribute.

```
Year|96|2|3
Make|CHEV|5|8
Model|Monte Carlo|10|20
~~Year|34|23|24~~
Feature|Red|42|44
Mileage|15,000|46|51
KEYWORD(Mileage)|miles|60|64
Price|14,990|84|89
~~Price|750|93|95~~
~~Mileage|1000|100|103~~
~~Make|CHEVROLET|120|128~~
PhoneNr|298-8090|130|137
```

Conclusion

The approaches given are useful in their domain. However, no general way of dealing with the construction of linguistic resources is given. A lot of heuristics and engineering is necessary for every single task.

References

additionally to the texts cited above:
Wordnet Homepage:
http://www.cogsci.princeton.edu/~wn/
(especially „Five Papers on Wordnet")

Fellbaum, C., (1998). WordNet, an electronical lexical database. Cambride, MA: MIT Press. (especially: Voorhees E. M.. Using WordNet for Text Retrieval.)