Automated Discovery of Telic Relations for WordNet

Marco De Boni
Suresh Manandhar

Background Terms

Glosses

– Every word and synset in WordNet contains a short (about one sentence long) description called glosses.

Telic Relations

- The goal / function of an object. The purpose an agent has in performing an act or the built-in function.

Background Terms (cont)

Telic Relations (cont)

- Some examples of telic relations,
  - The telic of "milk" might be "drink".
  - The telic of "wood" might be either "burn" (make a fire) or build (make furniture).

- Objects may have none, just one, or many telic relationships.

Goal of Article

Telic relations are talked about in WordNet (5 papers, pg. 18, 22), but never actually implemented.

The Goal of the article is to present an algorithm to automatically discover telic relations of words and synsets by looking at glosses (the descriptions of words).

General algorithm

For every noun in WordNet

1. Parse out a telic relation(s) from the glosses for the noun.
2. Since the telic word(s) will usually have many senses, find the appropriate synset to match up the noun to.

The Algorithm (Pt I: finding the telic word)

- By looking at special patterns (specific wording) within the glosses, telic words can be parsed out.
- The patterns used were

Pattern	Ex. Word	Gloss	telic
"… to TELIC_VERB by use of …"	Mammography	"a diagnostic procedure to detect breast tumors by use of X rays."	Detect breast tumors
"… used for Telic."	Tracing paper	"a semitransparent paper used for tracing drawings."	Trace drawings
"… used to Telic."	Cardiac glycoside	"Obtained from a number of plants and used to stimulate the heart …"	Stimulation of the Heart
"… use of … to Telic."	Trickery	"the use of tricks to deceive someone."	To deceive someone

The Algorithm (Pt I: finding the telic word)

- The patterns used were

Pattern	Ex. Word	Gloss	telic
"… used as … in TELIC_ING-VERB."	Plasticine	"… resembling clay; used as a substitute for clay or wax in modeling."	Modeling
"… used in TELIC_ING-VERB."	Seal oil	"… from seal blubber; used in making soap and dressing leather and as a lubricant."	Making soap, dressing leather and lubrification.
"… used in … as a TELIC."	Giant taro	"Large evergreen … used in wet warm regions as a stately ornamental."	Use as a stately ornamental.
"… for use as TELIC."	Houseboat	"a barge that is designed and equipped for use as a dwelling."	Dwelling.
"… for use in … TELIC_ING-VERB."	Wherry	"light rowboat for use in racing or for transporting goods and passengers in inland waters and harbors."	Racing, transportation of passengers.

Problems.

- Sometimes its hard to pick one specific noun phrase or verb phrase for the telic word.

- Over-generalization: When words like "be", "do", "make", "thing" were the telic relations, more specific words had to be found.

The Algorithm (pt II: finding the correct synset)

The telic word(s) of an object will most likely belong to more than one synset. The second part of the algorithm is to determine the appropriate synset for the object by looking at the distance (or difference) between different concepts.

There are many different approaches that can be use to find distance between concepts (for example wnconnect). The authors measured the relationship between two concepts with "Sematic distance."

Semantic Distance

Given a object w and its telic t, the semantic distance takes all glosses from each sense of t and compares it to the gloss of w (the object). The semantic distance is a sigmoid function that applies an number to each sense of t (the telic).

To calculate the actual sematic distance the authors look at all the words surrounding the glosses in t (the telic), and words in gloss w (the object).

Finding the Correct Synset ts

Given object w with gloss GW_w, it has a telic t with T representing all possible synsets (senses) of t. We want to find the correct synset, ts $\in$ T, by:

$$ts = Argmax_{ts' \in T}\ sd(\ GW_w,\ ts')^*$$

Where sd(a, s) calculates the semantic distance between sentence a and word w.

*I have no clue what exactly Argmax means

Calculating sd

In Order to find semantic distance sd for each ts' T, we take all the words in the gloss of ts' (call it set GT), plus all the words in the glosses of the hypernyms and hyponyms of ts' to the depth of 3, creating the set TS_{ts}. So

TS_{ts} = { w | w∈GT V
w∈hyperg(ts', 3) V
w∈hypog(ts', 3) }.

Calculating sd

Thus there are the sets

- GW_w = set of all words in gloss of object w.
- for every ts'$\in$T, TS = set all words in the gloss of ts' + hypernyms and hyponyms glosses (to depth of 3).

To reduce the above sets, the authors created a set of unimportant words ("the", "do", etc.) called the stop words, SW, so that

- $RGW_w = GW_w - SW$.
- RTS = TS – SW.

Calculating sd

Finally the algorithm compares each word in RGWw (reduced set of words in gloss for object w) and RTS (the reduced set of words in glosses of ts').

It compares relationship r by assigning weight depending on the type of relationship between two words (i.e. the strongest relationship is synonyms, hyper/hyponyms is next strongest, then satellites, etc.) thus calculating sd with

$$sd = |RTS| / ((\Sigma_{w \in RGW} r(w, RTS)) + 1)^*$$

*I assume that |RTS| = number of words in the set. Also the authors don't go into details about the r function, but if you assume the greater the relationship between w and RTS, the greater the weight, that would suggest that the greater the difference between two concepts the greater the sd.

The Algorithm in Review

For object word w, and set of stop (useless) words ST.

- Let set RGW = {gloss of w} – ST;
- For every parsing pattern
  Let word t = parsed out telic in gloss of w;
- If t is too general, reparse gloss of w for more specific t
- If t is too complex, go to next word.
- For every sense ts of t
  let set ts.RTS = {words in gloss ts + hyperg(ts, 3) + hypog(ts,3)} – ST;
  let semantic distance of ts, ts.sd = |RTS| / (($\Sigma_{w \in RGW}$ r(w, RTS)) + 1);
- Having assigned weights for each sense ts of t, create a link in WordNet between the word w and the synset of t with the lowest sd.

Success of Algorithm

2449 telic relationships were found relating to 1841 different synsets. It was estimated* that

- 77% of relationships were the actual correct telic.
- 1% of relationships were wrong.
- 9% of words had telic relationships that were too complex.
- The rest found useful but non-telic relationships.

*10% of the total relationships had to be manually checked.

Future work.

More manual review is required to verify that the algorithm actually worked.

Work is needed for when a single word is not enough to represent a telic relationship.

This is only one of many different semantic relations that can be implemented (Pustejovsky, 1995).