9B. Supertagging

Gerald Penn
Department of Computer Science, University of Toronto

Based upon slides by Michael Auli, Rober Hass and Aravind Joshi
WHY SUPERTAG?

- If lexical items have more description associated with them, parsing is easier
  - Only useful if the supertag space is not huge

- Straightforward to compile parse from accurate supertagging
  - But impossible if there are any supertag errors
    - We can account for *some* supertag errors
    - Don’t always want a full parse anyway
**What is Supertagging?**

- Systematic assignment of supertags

  Supertags are:

  - Statistically selected
    - Robust
    - Tends to work

  - Linguistically motivated
    - This makes sense
**What is supertagging?**

- Many supertags for each word
  - **Extended Domain of Locality**
    - Each lexical item has one supertag for every syntactic environment it appears in.
    - Inspiration comes from LTAG, lexicalized tree-adjointing grammars, in which *all* dependencies are localized.
  - Generally, agreement features such as number and tense, are not part of the supertag.
HOW TO SUPERTAG

“Alice opened her eyes and saw.”

Supertags:

- Verb
  - Transitive verb
  - Intransitive verb
  - Infinitive verb
  - ...

- Noun
  - Noun phrase (subject)
  - Nominal predicative
  - Nominal modifier
  - Nominal predicative subject extraction
  - ...
HOW TO SUPERTAG

“Alice opened her eyes and **saw**.”

- **Supertags**:
  - **Verb**
    - Transitive verb
    - Intransitive verb
    - Infinitive verb
    - ...
  - **Noun**
    - Noun phrase (subject)
    - Nominal predicative
    - Nominal modifier
    - Nominal predicative subject extraction
    - ...

```
S  VP  NP↓  saw  NP↓
```
HOW TO SUPERTAG

- A supertag can be ruled out for a given word in a given input string...
  - Left and/or right context is too long/short for the input
  - If the supertag contains other terminals not found in the input
HOW TO SUPERTAG

“Alice opened her eyes and saw.”

Supertags:

- Verb
  - Transitive verb
  - Intransitive verb
  - Infinitive verb
  - ...

- Noun
  - Noun phrase (subject)
  - Nominal predicative
  - Nominal modifier
  - Nominal predicative subject extraction
  - ...


HOW TO SUPERTAG

- This works fairly well
  - 50% average reduction in number of possible supertags
How to Supertag

...but there’s more to be done

- Good: average number of possible supertags per word reduced from 47 to 25
- Bad: average of 25 possible supertags per word
HOW TO SUPERTAG

- Disambiguation by unigrams?
  - Give each word its most frequent supertag after PoS tagging
    - ~75% accurate
      - Better results than one might expect given large number of possible supertags
      - Common words (determiners, etc.) usually correct
        - This helps accuracy
      - Back off to PoS for unknown words
        - Also usually correct
HOW TO SUPERTAG

- Disambiguation by n-grams?

\[
T = \text{argmax}_T \Pr(T_1, T_2, ..., T_N) \times \Pr(W_1, W_2, ..., W_N|T_1, T_2, ..., T_N)
\]

- We assume that subsequent words are independent

\[
\Pr(W_1, W_2, ..., W_N|T_1, T_2, ..., T_N) \approx \prod_{i=1}^{N} \Pr(W_i|T_i)
\]

- Trigrams plus Good-Turing smoothing
  - Accuracy around 90%
    - Versus 75% from unigrams
  - Contextual information more important than lexical
    - Reversal of trend for PoS tagging
However...

- Correctly supertagged text yields a 30X parsing speedup
  - But even one mistake can cause parsing to fail completely
    - This is rather likely

- Solution: n-best supertags?
  - When n=3, we get up to 96% accuracy...
    - Not bad at all for such a simple method
    - 425 lexical categories (PTB-CFG: ~50)
    - 12 combinatory rules (PTB-CFG: > 500,000)