Bringing Structurally Rich Language Models into Service

Johanna Björklund

Natural and Formal Languages
Dept. Computing Science, Umeå University

23 februari 2015
Elevator pitch
What is this all about?

The way natural language is analysed by e.g. Google, Siri, etc. has become a limitation.

In particular, there is no support for long-distance or hierarchical dependencies.

Leads to mistakes such as

She rowed between Yellowknife and Benchoko in a car

Richer language-models have more difficult computational problems.

We use a set of optimisation techniques to speed up existing algorithms for central problems.
What can it be used for?

Better quality **machine translation**

Representative and readable **summarization**

**Knowledge mining** from unstructured text

Higher accuracy and improved robustness in **speech transcription**

Improved **sentiment analysis** and **authorship attribution**
How does it relate to other research?

Challenge to develop optimisation techniques that generalize.

Focus shift from worst-case to average-case complexity.

Combines mathematical reasoning, engineering methodology, and empirical approaches.
Once More, with Feeling
Language models

Most language-technological applications include a language model. This assigns likelihood to natural language sentences.

- She turned me into a newt
- This is an ex-parrot
- Spam spam spam spam spam wonderful spam
Reigning model is \( N \)-grams

\( N \) is some natural number, in practice \( \leq 5 \)

Sentences are segmented into \( N \)-word sequences

Works surprisingly well, but is not the full answer

- She rowed between Yellowknife and Benchoko in a canoe
- She rowed between Yellowknife and Benchoko in a car

Size of model grows \textit{exponentially} in \( N \), so it fails to scale
Structural models (1/2)

Economic news had little effect on financial markets.
 Structural models (2/2)

Minor flora of more structured models, e.g.
  • grammars for directed acyclic graphs,
  • tree adjoining grammars (TAGs), and
  • hybrid grammars
are under consideration.

Important related problems
  • inference,
  • training, and
  • parsing.

Worst-case complexities can be pretty horrible, c.f. $O(n^6)$ for TAG parsing compared to $O(n \log n)$ for $N$-grams.
Optimisation techniques

Lexicalisation and linearisation. Model is restricted to tie computation length closer to input size.

Iterative refinement. Start out with a coarse approximation, and refine until 'good enough'.

Serialization with pruning. Break non-deterministic computations into several steps, and prune away unpromising branches.

Hyper-minimisation. Allow 'small' mistakes to be able to work with a more compact language model.
Preliminary results

Linearisation and lexicalisation. Evaluation on ASR task with trigram as baseline. Structural-agnostic model has a 43% error rate, structurally-aware model has a 3% error rate.

Serialization with pruning. Efficient algorithm for computing best analyses of context-free grammars

Application: authorship attribution. Early results suggests that syntactical patterns could be more useful than vocabulary to separate between authors writing on different topics.
Impact

Better quality machine translation

Representative and readable summarization

Knowledge mining from unstructured text

Higher accuracy and improved robustness in speech transcription

Improved sentiment analysis and authorship attribution
Thank you!