UNIVERSITY OF TAMPERE

SIS
SCHOOL OF INFORMATION SCIENCES

Text summarization using GF?

Jyrki Nummenmaa

GF Summer School 27.8.2013

Text Summarization

- Like making an abstract
  - Most of us have done that
  - Even if it feels like mechanical work sometimes, it seems hard to automate.
- Hand-made summaries can find creative ways to say the same things using different expressions.
- We are now considering automatic summarization, where this is not quite so.

Single-source summarization

- When there is a single source, the aim is to somehow indentify and prune the less important parts (or identify and extract the more important parts. E.g:

  This boring fish is expensive

- Why? How? Is this correct?

How to summarize?

- Hierarchical structure
- Events and their relationships (causal, temporal, etc)
- Content analysis and scoring
- Heuristics
- Grammatical analysis?
  Is (This (Qkind Boring Fish)) Expensive
- Let’s have a look at some more realistic example inputs.

Problems

- Evaluation of the result is a bit problematic
- If only existing sentences are chosen for the output, then it is limited what the method can generate.
- Some methods can identify important concepts well, but have a hard time generating sentences.

Multisource summarization

- From multiple sources, the analysis is based on the commonality of the content in different sources.

This boring fish is expensive

This fish is expensive

- Seems like an easy choice…

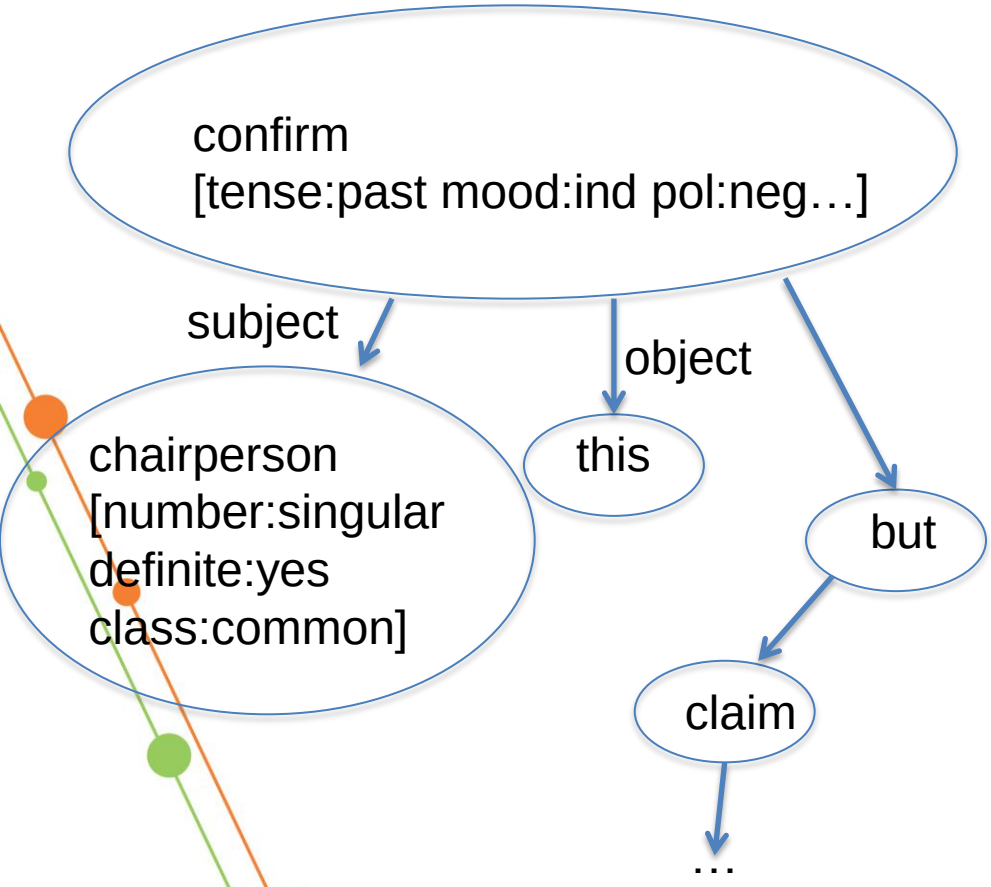
Events and concepts

- Can be ordered and ranked numerically.
- Unless already existing sentences are chosen, the problem of sentence generation remains.
  - Needless to say, if the sentences are grammatically incorrect, the summary looks stupid.

Dependency trees

- Give structure to the information, which helps in both
  - Selecting the information to the summary, and
  - Generating the sentences.
- Sentence-generation remains problematic.

Dependency trees



- Give structure to the information, which helps in both
  - Selecting the information to the summary, and
  - Generating the sentences.
- Sentence-generation remains problematic.

Sentence fusion

- Identifying important sentence parts and then forming sentences from them.
  - Sentence-generation is still problematic.

Grammatical Framework?

- Potential idea that lacks an implementation:
  - Parse the texts with the robust parser.
  - Take all the parse trees.
  - Find commonalities and merge / eliminate sub-trees that do not appear elsewhere / find containing subtrees
  - Linearize
- Eventually statistics would be useful (hybrid method), but it will be interesting to see, what (if any…) can be achieved this or some other way using GF

Trivial example

1. This fish is expensive
   This boring fish is expensive
2. Is (This Fish) Expensive
   Is (This (Qkind Boring Fish)) Expensive
3. Is (This Fish) Expensive
4. This fish is expensive

Inputs

- Available from me and from the Internet, e.g. the “Newsblaster”
- Some I have parsed with the robust GF parser
  - the experiments hopefully help to make it even more robust…
- Some preprocessing:
  - .,“” ,– –, ( ), ?, ;, smallcaps to start sentences, remove links, subtitles, etc.

The processing

- E.g. use Haskell to
  - Read in the abstract grammar expressions,
  - Manipulate them to summarize
  - Linearize the summary
- Why not with something else…

Thanks!

Questions?

Contact:
jyrki.nummenmaa@uta.fi