Automatic semantic clustering of text corpus contexts

SemWeb seminar

Pavel Rychlý

Faculty of Informatics
Masaryk University
Brno, Czech Republic

5–7 October, 2006

Can computers understand free text?

- computer understanding: symbolic manipulation

Can computers understand free text?

- computer understanding: symbolic manipulation
- needs for *elements/symbols*

Can computers understand free text?

- computer understanding: symbolic manipulation
- needs for *elements/symbols*
    - character

Can computers understand free text?

- computer understanding: symbolic manipulation
- needs for *elements/symbols*
    - character
    - word

Can computers understand free text?

- computer understanding: symbolic manipulation
- needs for *elements/symbols*
  - character
  - word
  - lemma

Can computers understand free text?

- computer understanding: symbolic manipulation
- needs for *elements/symbols*
    - character
    - word
    - lemma
    - meaning

word as an element for understanding

- a word without context – no meaning

word as an element for understanding

- a word without context – no meaning
- a word in different contexts – different meanings

word as an element for understanding

- a word without context – no meaning
- a word in different contexts – different meanings
- words in *similar* contexts – OK

word as an element for understanding

- a word without context – no meaning
- a word in different contexts – different meanings
- words in *similar* contexts – OK
- what is context?

What is context

- Which words?:

What is context

- Which words?:
  - next word

What is context

- Which words?:
  - next word
  - last word

What is context

- Which words?:
  - next word
  - last word
  - window, +1 to +5

What is context

- Which words?:
  - next word
  - last word
  - window, +1 to +5
  - window, -5 to -1

What is context

- Which words?:
  - next word
  - last word
  - window, +1 to +5
  - window, -5 to -1
  - use only typical

What is context

- Which words?:
  - next word
  - last word
  - window, +1 to +5
  - window, -5 to -1
  - use only typical
- How sorted?

What is context

- Which words?:
  - next word
  - last word
  - window, +1 to +5
  - window, -5 to -1
  - use only typical
- How sorted?
- most common collocates – but for most nouns it's the

What is context

- Which words?:
  - next word
  - last word
  - window, +1 to +5
  - window, -5 to -1
  - use only typical
- How sorted?
- most common collocates – but for most nouns it’s the
- most salient collocates – how to measure salience?

Word Sketch

A corpus-derived one-page summary of a word's grammatical and collocational behaviour ▸ try online

Word Sketch

How to create one

- Large well-balanced corpus

Word Sketch

How to create one

- Large well-balanced corpus
- Parse to find subjects, objects, heads, modifiers etc

Word Sketch

How to create one

- Large well-balanced corpus
- Parse to find subjects, objects, heads, modifiers etc
- One list for each grammatical relation

Word Sketch

How to create one

- Large well-balanced corpus
- Parse to find subjects, objects, heads, modifiers etc
- One list for each grammatical relation
- Statistics to sort each list

The Sketch Engine

- Input:

The Sketch Engine

- Input:
  - any corpus, any language

The Sketch Engine

- Input:
  - any corpus, any language
  - Lemmatised, part-of-speech tagged

The Sketch Engine

- Input:
  - any corpus, any language
  - Lemmatised, part-of-speech tagged
  - specification of grammatical relations

The Sketch Engine

- Input:
  - any corpus, any language
  - Lemmatised, part-of-speech tagged
  - specification of grammatical relations
- Word sketches integrated with

The Sketch Engine

- Input:
  - any corpus, any language
  - Lemmatised, part-of-speech tagged
  - specification of grammatical relations
- Word sketches integrated with
- Corpus query system

The Sketch Engine

- Input:
  - any corpus, any language
  - Lemmatised, part-of-speech tagged
  - specification of grammatical relations
- Word sketches integrated with
- Corpus query system
  - Supports complex searching, sorting etc

The Sketch Engine

- Input:
  - any corpus, any language
  - Lemmatised, part-of-speech tagged
  - specification of grammatical relations
- Word sketches integrated with
- Corpus query system
  - Supports complex searching, sorting etc
  - IMS-Stuttgart formalism (also for corpus input)

The Sketch Engine

- Input:
  - any corpus, any language
  - Lemmatised, part-of-speech tagged
  - specification of grammatical relations
- Word sketches integrated with
- Corpus query system
  - Supports complex searching, sorting etc
  - IMS-Stuttgart formalism (also for corpus input)
  - Corpus searches and grammar writing

Grammatical Relations Definition

- plain text file

Grammatical Relations Definition

- plain text file
- a set of queries for each GR

Grammatical Relations Definition

- plain text file
- a set of queries for each GR
- queries contain labels for keyword and collocate

Grammatical Relations Definition

- plain text file
- a set of queries for each GR
- queries contain labels for keyword and collocate
- processing options

GR Definition Examples

```
# ‘adverb’ gramrel definition
=adverb
   1:[] 2:"AV."
   2:"AV." 1:[]

# ‘and/or’ gramrel definition
=and/or
*SYMMETRIC
   1:[] [word="and"|word="or"] 2:[] & 1.tag = 2.tag
```

GR Definition Examples

```
# ‘modifier’ and ‘modify’ gramrels definition
*DUAL
=modifier/modify
   2:"AJ." 1:"N.."

*UNARY
=wh_word
1:[] [tag="AVQ"|tag="DTQ"|tag="PNQ"]

*TRINARY
=pp_%s
1:[tag="N.."|tag="AJ."] 3:"PR." 2:"N.."
```

Association score

- counting $(word_1, gramrel, word_2)$

Association score

- counting $(word_1, gramrel, word_2)$
- $AScore(w_1, R, w_2) =$
  $\log \frac{||w_1,R,w_2|| \cdot ||*,*,*||}{||w_1,R,*|| \cdot ||*,*,w_2||} \cdot \log(||w_1, R, w_2|| + 1)$

Similarity score

- comparing w_1 and w_2's word sketches

$$Dist(w_1, w_2) = \frac{\sum_{(tup_i, tup_j) \in \{tup_{w_1} \cap tups_{w_2}\}} AS_i + AS_j - (AS_i - AS_j)^2/50}{\sum_{tup_i \in \{tup_{w_1} \cup tup_{w_2}\}} AS_i}$$

Similarity score

- comparing w_1 and w_2's word sketches
- only important context

$$Dist(w_1, w_2) = \frac{\sum_{(tup_i, tup_j) \in \{tup_{w_1} \cap tups_{w_2}\}} AS_i + AS_j - (AS_i - AS_j)^2/50}{\sum_{tup_i \in \{tup_{w_1} \cup tup_{w_2}\}} AS_i}$$

Similarity score

- comparing w_1 and w_2's word sketches
- only important context
- how much overlaps

$$Dist(w_1, w_2) = \frac{\sum_{(tup_i, tup_j) \in \{tup_{w_1} \cap tups_{w_2}\}} AS_i + AS_j - (AS_i - AS_j)^2/50}{\sum_{tup_i \in \{tup_{w_1} \cup tup_{w_2}\}} AS_i}$$

Similarity score

- comparing w_1 and w_2's word sketches
- only important context
- how much overlaps
- counting $(word_1, (gramrel, word_i))$ and $(word_2, (gramrel, word_i))$

$$Dist(w_1, w_2) = \frac{\sum_{(tup_i, tup_j) \in \{tup_{w_1} \cap tups_{w_2}\}} AS_i + AS_j - (AS_i - AS_j)^2/50}{\sum_{tup_i \in \{tup_{w_1} \cup tup_{w_2}\}} AS_i}$$

Thesaurus entry and collocates clustering

- bottom-up hierarchical clustering

Thesaurus entry and collocates clustering

- bottom-up hierarchical clustering
- select more items

Thesaurus entry and collocates clustering

- bottom-up hierarchical clustering
- select more items
- group singletons with highest similarity

Thesaurus entry and collocates clustering

- bottom-up hierarchical clustering
- select more items
- group singletons with highest similarity
- drop clusters over fixed limit