(Computational) Lexical Semantics

MLP Course, winter term 11/12

based on chapters 19/12, Jurafsky and Martin

December 21, 2011

Outline

1. Lexical Semantics (Chapter 19, J+M)
   - Word senses
   - Relations between word senses
   - WordNet
   - Lexical semantics of verbs
   - Challenges

2. Computational Lexical Semantics (Chapter 20, J+M)
   - Word Sense Disambiguation
   - Word Similarity
   - Semantic Roles Labeling
   - Towards tracking semantic change by visual analytics (Rohrdantz et al 2011)

Outline

1. Lexical Semantics (Chapter 19, J+M)
   - Word senses
   - Relations between word senses
   - WordNet
   - Lexical semantics of verbs
   - Challenges

2. Computational Lexical Semantics (Chapter 20, J+M)
   - Word Sense Disambiguation
   - Word Similarity
   - Semantic Roles Labeling
   - Towards tracking semantic change by visual analytics (Rohrdantz et al 2011)

Word senses

'the bow'

"The **bow** *should be tall enough to prevent water from washing over the ship."*

"The **bow** *consists of a specially shaped stick and a ribbon stretched between its ends and is used to stroke the strings and create sound."*

"Robin Hood used **bow** *and arrow to fight the rich."*

"The level and duration of the **bow** *depends on status, age and other factors."*

Word senses

'the bow'

"The ***bow*** *should be tall enough to prevent water from washing over the ship."*

- a ship's bow

"The ***bow*** *consists of a specially shaped stick and a ribbon stretched between its ends and is used to stroke the strings and create sound."*

- the bow of a musical instrument

"Robin Hood used ***bow*** *and arrow to fight the rich."*

- the bow as a weapon

"The level and duration of the ***bow*** *depends on status, age and other factors."*

- the bow as a movement

Word senses

'the bow'

"The ***bow*** *should be tall enough to prevent water from washing over the ship."*

- a ship's bow

"The ***bow*** *consists of a specially shaped stick and a ribbon stretched between its ends and is used to stroke the strings and create sound."*

- the bow of a musical instrument

"Robin Hood used ***bow*** *and arrow to fight the rich."*

- the bow as a weapon

"The level and duration of the ***bow*** *depends on status, age and other factors."*

- the bow as a movement

- The noun *bow* has at least **four senses**

Word Senses

- one word, but its senses are completely unrelated
  - e.g. *bank*
  - homonyms → homonymy
- one word, its senses are semantically related
  - *bow* as in weapon and part of a musical instrument
  - polysems → polysemy

→ gradual distinction between homonymy and polysemy

Word Senses

- one word, but its senses are completely unrelated
  - e.g. *bank*
  - homonyms → homonymy
- one word, its senses are semantically related
  - *bow* as in weapon and part of a musical instrument
  - polysems → polysemy

→ gradual distinction between homonymy and polysemy

- one aspect of a concept refers to another aspect of that concept
  - e.g. usage of *White House* when referred to the administration with offices in the White house
  - metonymy

Relations between word senses

- two words with (almost) identical senses
  - *couch/sofa*, *to vomit/to throw up*
  - synonymy
  - more formally: *two words are synonymous if they are substitutable without changing the truth conditions of the sentence*
- two words with opposed senses
  - *short/long*, *rise/fall*
  - antonymy

Relations between word senses

- one sense is more specific than another sense
  - *hyponymy*
- one sense is less specific than another sense
  - hypernymy

hypernym	vehicle	fruit	furniture
hyponym	car	mango	chair

- senses are related by a part-whole relation
  - *leg/chair*, *wheel/car*
  - "part" = *leg* = meronym, "whole" = *chair* = holonym

→ these concepts are the building blocks of a taxonomy, i.e. a tree-like structure of senses

WordNet

- the most commonly used lexical resource for English words is WordNet (Fellbaum, 1998)
- based on the relations of senses as just discussed
- three separate databases for nouns, verbs and adjectives/adverbs
- WordNet 3.0 has 117097 nouns, 11488 verbs, 22141 adjectives and 4601 adverbs

Demo

Lexical semantics of verbs

Representation of an event in a neo-Davidsonian way:

Jane broke the window.

$\exists$e,x,y Breaking(e) $\wedge$ Jane(x) $\wedge$ window(y) $\wedge$

Lexical semantics of verbs

Representation of an event in a neo-Davidsonian way:

Jane broke the window.

$\exists$e,x,y Breaking(e) $\wedge$ Jane(x) $\wedge$ window(y) $\wedge$
Breaker(e,x) $\wedge$ BrokenThing(e,y)

Lexical semantics of verbs

Representation of an event in a neo-Davidsonian way:

Jane broke the window.

$\exists$e,x,y Breaking(e) $\wedge$ Jane(x) $\wedge$ window(y) $\wedge$
Breaker(e,x) $\wedge$ BrokenThing(e,y)

- Breaker and BrokenThing are **deep** roles and are specific to each event
- BUT: in order to build computational systems we need to have a more general classification of arguments
- different approaches:
  - thematic roles (Fillmore 1968 and Gruber 1965)
  - proto roles as in PropBank
  - frame-specific roles as in FrameNet

Lexical semantics of verbs

Thematic roles (Fillmore 1968 and Gruber 1965)

Thematic Role	**Definition**
AGENT	The volitional causer of an event
EXPERIENCER	The experiencer of an event
FORCE	The non-volitional causer of the event
THEME	The participant most directly affected by an event
RESULT	The end product of an event
CONTENT	The proposition or content of a propositional event
INSTRUMENT	The instrument used in an event
BENEFICIARY	The beneficiary of an event
SOURCE	The origin of the objet of a transfer event
GOAL	The destination of an object of a transfer event

Lexical semantics of verbs

Representation of verb arguments with thematic roles:

Jane broke the window.

Lexical semantics of verbs

Representation of verb arguments with thematic roles:

Jane broke the window.

Jane = Agent, the window = Theme

Lexical semantics of verbs

Representation of verb arguments with thematic roles:

Jane broke the window.

Jane = Agent, the window = Theme

Jane broke the window with a rock.

Lexical semantics of verbs

Representation of verb arguments with thematic roles:

Jane broke the window.

Jane = Agent, the window = Theme

Jane broke the window with a rock.

Jane = Agent, the window = Theme, the rock = Instrument

Lexical semantics of verbs

Representation of verb arguments with thematic roles:

Jane broke the window.

Jane = Agent, the window = Theme

Jane broke the window with a rock.

Jane = Agent, the window = Theme, the rock = Instrument

The window was broken by Jane.

Lexical semantics of verbs

Representation of verb arguments with thematic roles:

Jane broke the window.

Jane = Agent, the window = Theme

Jane broke the window with a rock.

Jane = Agent, the window = Theme, the rock = Instrument

The window was broken by Jane.

the window = Theme, Jane = Agent

- Possible arguments of *to break*: AGENT, THEME, INSTRUMENT

Lexical semantics of verbs

But verbs can vary according to which thematic roles they assign in what position:

(1) a. Jane broke the window.

b. The window broke.

(2) a. Jane cut the cake.

b. *The cake cut. Conative alternation

Lexical semantics of verbs

But verbs can vary according to which thematic roles they assign in what position:

(4) a. Jane broke the window.
    b. The window broke.

(5) a. Jane cut the cake.
    b. *The cake cut. Conative alternation

(6) a. Jane gave the book to James.
    b. Jane gave James the book. Dative alternation

- Levin (1993) is a reference book that lists all verb alternations for English and detects semantic classes of verbs based on their syntactic behavior → basis for the English Verbnet (Demo)

Lexical semantics of verbs

The Proposition Bank (PropBank)

- the PennTreebank annotated with semantic roles
- semantic roles are defined with respect to an individual verb sense
- roles in PropBank are numbered rather than labeled, e.g. Arg0, Arg1 etc.

Lexical semantics of verbs

The Proposition Bank (PropBank)

- the PennTreebank annotated with semantic roles
- semantic roles are defined with respect to an individual verb sense
- roles in PropBank are numbered rather than labeled, e.g. Arg0, Arg1 etc.

agree.01	
Agr0:	Agreer
Agr1:	Proposition
Agr2:	Other entity agreeing
Ex1:	[$_{Agr0}$ The group] *agreed* [$_{Agr1}$ it wouldn't make an offer unless it had Georgia Gulf's consent].
Ex2:	[$_{ArgM-TMP}$ Usually] [$_{Arg0}$ John] *agrees* [$_{Arg2}$ with Mary] [$_{Arg1}$ on everything].

Lexical semantics of verbs

Problems with PropBank

$\surd$ [$_{Agr0}$ The group] *agreed* [$_{Agr1}$ it wouldn't make an offer unless it had Georgia Gulf's consent].

$\surd$ [$_{ArgM-TMP}$ Usually] [$_{Arg0}$ John] *agrees* [$_{Arg2}$ with Mary] [$_{Arg1}$ on everything].

Lexical semantics of verbs

Problems with PropBank

√ [$_{Agr0}$ The group] *agreed* [$_{Agr1}$ it wouldn't make an offer unless it had Georgia Gulf's consent].
√ [$_{ArgM-TMP}$ Usually] [$_{Arg0}$ John] *agrees* [$_{Arg2}$ with Mary] [$_{Arg1}$ on everything].

? [$_{ArgM-TMP}$ Usually] [$_{Arg0}$ John] *consents* [$_{Arg2}$ with Mary] [$_{Arg1}$ on everything].
? There is an agreement of [$_{Arg0}$ John] with [$_{Arg2}$ with Mary].

We would like to represent these roles in a uniform way, across different verbs and also across nouns and verbs → FrameNet

Lexical semantics of verbs

FrameNet

- semantic role labeling project that attempts to address the problems of thematic roles and PropBank (Baker et al. 1998, Lowe et al. 1997 and Ruppenhofer et al. 2006)
- verbs are grouped in frames where specific roles hold
- e.g. frame *make_agreement_on_action*

Demo

Challenges

Two main challenges in the computational treatment of lexical semantics:

- selectional restrictions
  - semantic constraint that the verb imposes on the concepts that are allowed to fill its argument structure
- metaphors
  - relation between two completely different domains of meaning - generating an independent meaning

Challenges

Selectional restrictions:

(7) a. I want to eat Malaysian food.

b. I want to eat somewhere.

How do we know that *somewhere* is not the direct object of the sentence?

- intransitive and transitive version of *to eat*
- the direct object of *to eat* must be an edible entity
- *somewhere* is a location and not edible

Challenges

(8) a. Does American Airlines still serve a hot meal?

b. Does American Airlines still serve Denver?

Senses of serve:

- cooking/providing food
- providing a commercial service
- and probably other senses, too

$\rightarrow$ the set of concepts needed to represent selectional restrictions is almost open-ended

$\rightarrow$ no resource available that encodes a full range of these concepts (does a finite set of these concepts exist at all?)

Challenges

Can we get around the problem of selectional restrictions?

1. Usage of WordNet?
   - for the case of *to eat* we could refer to the synset *food, nutrient* for its direct object
   - but then we also need to account for cases like *I ate rabbit the other day* item include the synset *animal* as well?
2. Decomposing the meaning of words into their primitive semantic elements?
   - What would these elements be for *cow, bull, calf*?

Challenges

A further problem for computers: metaphors

(9) It doesn't scare Microsoft that Apple's new IPad is out.

- here, the company is viewed as a person that can experience fear
- problem for the computer: when is an expression metaphorically used and when is it ill-formed?
  - ?Apple is scared of mice.

Quick recap

- Relations between word senses:
  - synonymy
  - antonymy
  - hyponymy/hypernymy
  - meronymy
- verb lexical semantics
  - thematic roles
  - proto-roles
  - frame roles

Outline

1. Lexical Semantics (Chapter 19, J+M)
   - Word senses
   - Relations between word senses
   - WordNet
   - Lexical semantics of verbs
   - Challenges

2. Computational Lexical Semantics (Chapter 20, J+M)
   - Word Sense Disambiguation
   - Word Similarity
   - Semantic Roles Labeling
   - Towards tracking semantic change by visual analytics (Rohrdantz et al 2011)

Word Sense Disambiguation (WSD)

Two main approaches:

1. lexical sample approach
   - a small pre-selected set of target words to be disambiguated
   - set of senses for each word from a lexicon
   - corpus instances of the target words are hand-labelled with the correct senses
     - e.g. *line-hard-serve* corpus (Leacock et al. 1993), *interest* corpus (Bruce and Wiebe 1994) and SENSEVAL corpora
   - classifier systems are trained on these instances
   - unlabeled instances are then tagged with the classifier

Word Sense Disambiguation (WSD)

Two main approaches:

1. lexical sample approach
    - a small pre-selected set of target words to be disambiguated
    - set of senses for each word from a lexicon
    - corpus instances of the target words are hand-labelled with the correct senses
        - e.g. *line-hard-serve* corpus (Leacock et al. 1993), *interest* corpus (Bruce and Wiebe 1994) and SENSEVAL corpora
    - classifier systems are trained on these instances
    - unlabeled instances are then tagged with the classifier
2. all-words approach
    - a system is given a text and a lexicon with senses of the words of the text
        - e.g. SemCor (Miller et al. 1993, Landes et al. 1998) and SENSEVAL-3 (Palmer et al. 2001)
    - then every content word of the text is disambiguated

Word Sense Disambiguation (WSD)

1. Supervised learning:
    1. extraction of features that are predictive of word senses
        - collocational features: position-specific relation to the target word
        - bag-of-words features: unordered set of words, exact position is ignored

Word Sense Disambiguation (WSD)

1. Supervised learning:
   1. extraction of features that are predictive of word senses
      - collocational features: position-specific relation to the target word
      - bag-of-words features: unordered set of words, exact position is ignored

An electric guitar and bass player stand off to one side, just as a sort of nod to gringo expectation perhaps.

Collocational feature vector with target word w_i:

$[\mathrm{w}_{i-2}, \mathrm{POS}_{i-2}, \mathrm{w}_{i-1}, \mathrm{POS}_{i-1}, \mathrm{w}_{i+1}, \mathrm{POS}_{i+1}, \mathrm{w}_{i+2}, \mathrm{POS}_{i+2}]$

Word Sense Disambiguation (WSD)

1. Supervised learning:
    1. extraction of features that are predictive of word senses
        - collocational features: position-specific relation to the target word
        - bag-of-words features: unordered set of words, exact position is ignored

An electric guitar and bass player stand off to one side, just as a sort of nod to gringo expectation perhaps.

Collocational feature vector with target word w_i:

$[\mathrm{w}_{i-2}, \mathrm{POS}_{i-2}, \mathrm{w}_{i-1}, \mathrm{POS}_{i-1}, \mathrm{w}_{i+1}, \mathrm{POS}_{i+1}, \mathrm{w}_{i+2}, \mathrm{POS}_{i+2}]$

[guitar, NN, and, CC, player, NN, stand, VB]

Word Sense Disambiguation (WSD)

1. Supervised learning:

An electric guitar and bass player stand off to one side, just as a sort of nod to gringo expectation perhaps.

Vocabulary vector of the 10 most frequent content words in *bass* sentences:
[*fishing, sound, player, fly, rod, double, runs, playing, guitar, band*]

Bag-of-words feature vector with binary features:

Word Sense Disambiguation (WSD)

1. Supervised learning:

An electric guitar and bass player stand off to one side, just as a sort of nod to gringo expectation perhaps.

Vocabulary vector of the 10 most frequent content words in *bass* sentences: [*fishing, sound, player, fly, rod, double, runs, playing, guitar, band*]

Bag-of-words feature vector with binary features:

[0,0,1,0,0,0,0,0,1,0]

Word Sense Disambiguation (WSD)

1. Supervised learning:

An electric guitar and bass player stand off to one side, just as a sort of nod to gringo expectation perhaps.

Vocabulary vector of the 10 most frequent content words in *bass* sentences: [*fishing, sound, player, fly, rod, double, runs, playing, guitar, band*]

Bag-of-words feature vector with binary features:

[0,0,1,0,0,0,0,0,1,0]

These vectors are then input to machine learning algorithms.

Word Sense Disambiguation (WSD)

Naive Bayes classifier:

- $\hat{s} = \text{argmax}\ \mathrm{P}(\mathrm{s}_i) \prod_{j=1}^{n} \mathrm{P}(\mathrm{f}_j|\mathrm{s}_i)$
- training a naive Bayes classifier means estimating each of these probabilities
- $\mathrm{P}(\mathrm{s}_i) = \frac{count(s_i, w_j)}{count(w_j)}$ = prior probability of each sense
  - counting the number of times sense s_i occurs, divided by the total number of target word w_j
  - If the target word *bass* appears 150 times in the corpus and it has sense *bass*[1] in 60 cases, what is the prior probability of the sense?

Word Sense Disambiguation (WSD)

Naive Bayes classifier:

- $\hat{s} = \text{argmax } P(s_i) \prod_{j=1}^{n} P(f_j|s_i)$
- training a naive Bayes classifier means estimating each of these probabilities
- $P(s_i) = \frac{count(s_i, w_j)}{count(w_j)}$ = prior probability of each sense
  - counting the number of times sense s_i occurs, divided by the total number of target word w_j
  - If the target word *bass* appears 150 times in the corpus and it has sense $bass^1$ in 60 cases, what is the prior probability of the sense?
- $P(f_j|s) = \frac{count(f_j, s)}{count(s)}$ = individual feature probabilities
  - If a feature such as $[w_{i-2} = \text{guitar}]$ occurs three times for sense $bass^1$, and sense $bass^1$ occurs 60 times in the corpus, what is its individual feature probability?

Word Sense Disambiguation (WSD)

Naive Bayes classifier:

- $\hat{s} = \text{argmax } P(s_i) \prod_{j=1}^{n} P(f_j | \, s_i)$
- putting in the values computed before for
  - $P(s_i) = \frac{count(s_i, w_j)}{count(w_j)}$ = prior probability of each sense
  - $P(f_j|s) = \frac{count(f_j, s)}{count(s)}$ = individual feature probabilities
- What is the probability of guitar occurring with sense $bass^1$?

Word Sense Disambiguation (WSD)

Evaluation of WSD systems:

- a WSD system can be evaluated with respect to **sense accuracy**
  - the percentage of words that are tagged identically to the hand-labeled sense tags in the test set
- usually compared to two measures:
  - baseline
    - e.g. simply take the most frequent sense for each word
  - ceiling
    - e.g. human inter-annotator agreement

Word Sense Disambiguation (WSD)

2. Dictionary and Thesaurus Methods

- The Lesk algorithm: family of algorithms for dictionary-based sense disambiguation
- Simplified Lesk algorithm (Kilgarriff and Rosenzweig 2000):
  - which sense gloss shares the most words with the target word's neighbourhood?

Word Sense Disambiguation (WSD)

2. Dictionary and Thesaurus Methods

- The Lesk algorithm: family of algorithms for dictionary-based sense disambiguation
- Simplified Lesk algorithm (Kilgarriff and Rosenzweig 2000):
  - which sense gloss shares the most words with the target word's neighbourhood?

The bank can guarantee deposits that will eventually cover future tuition costs because it invests in adjustable-rate mortgage securities.

bank^1	Gloss:	a financial institution that accepts deposits and channels the money into lending activities
bank^2	Gloss:	sloping land (especially the slope beside a body of water

Which sense is taken?

Word Sense Disambiguation (WSD)

2. Dictionary and Thesaurus Methods

- Original Lesk algorithm (Lesk 1986):
  - the gloss of the target word is compared to the glosses of the surrounding words
  - the sense with the most overlapping words is chosen

Word Sense Disambiguation (WSD)

2. Dictionary and Thesaurus Methods

- Original Lesk algorithm (Lesk 1986):
  - the gloss of the target word is compared to the glosses of the surrounding words
  - the sense with the most overlapping words is chosen

pine cone

pine^1	Gloss:	kinds of evergreen trees with needle-shaped leaves
pine^2	Gloss:	waste away through sorrow or illness
cone^1	Gloss:	solid body which narrows to a point
cone^2	Gloss:	something of this shape whether solid or hollow
cone^3	Gloss:	fruit of certain evergreen trees

Which sense is taken?

Word Sense Disambiguation (WSD)

The caveat of large hand-built resources

- both the supervised approach and the dictionary-based approach require large amounts of labeled data
- what can be done if these resources are not available?

$\rightarrow$ e.g. Yarovsky algorithm (1995)

  - small seedset of labeled instances of each sense and a much larger unlabeled corpus
  - first training of an initial classifier on the seedset
  - then parsing of the unlabeled data with this classifier
  - selection of the most confident labeled instance and addition to the training set
  - with each iteration, the training set grows and the unlabeled corpus shrinks

Word Similarity

- to compute word similarity is useful for many natural language applications

Word Similarity

- to compute word similarity is useful for many natural language applications
    - machine translation
    - information retrieval
    - question answering
    - text summarization

Word Similarity

- to compute word similarity is useful for many natural language applications
  - machine translation
  - information retrieval
  - question answering
  - text summarization
- two classes of algorithms: **thesaurus-based algorithms** and **distributional algorithms**

Word Similarity

1. Thesaurus-based algorithms:

- usage of the structure of a thesaurus to define word similarity
- word similarity $\neq$ word relatedness
  - word relatedness characterizes a larger set of potential relationship between words
  - e.g. antonyms are related but not similar
- Path-length-based similarity: measuring the edges between two concepts

$$\mathrm{sim}_{path}(c_1, c_2) = \mathrm{pathlen}(c_1, c_2)$$

- Log transform of path-length-based similarity

$$\mathrm{sim}_{path}(c_1, c_2) = -\log \mathrm{pathlen}(c_1, c_2)$$

Word Similarity

- problem with path-length algorithms:
  - assumption that each link in the thesaurus represents a uniform distance

$\rightarrow$ information-content word-similarity algorithms (following Resnik 1995)
  - the lower a concept in a hierarchy, the lower its probability
  - *P(c)* is the probability that a randomly selected word in a corpus is an instance of concept *c*
  - P(root) = 1 (any word is subsumed by the root concept)

$$\mathrm{P(c)} = \frac{\sum_{w \epsilon words(c)} count(w)}{N}$$

Word Similarity

- two additional definitions are needed:
  - informaton concent (IC) of a concept: IC(c) = - log P(c)
  - lowest common subsumer (LCS) of two concepts: LCS(c_1, c_2)
    - the lowest node in the hierarchy that subsumes (is a hypernym of) both c_1 and c_2.
- Resnik similarity measure:

$$\text{sim}_{resnik}(c_1, c_2) = -\log P(\text{LCS}(c_1, c_2))$$

→ information content of the lowest common subsumer of the two nodes

Word Similarity

2. Distributional Algorithms

- Intuition: the meaning of a word is related to the distribution of words around it
  - "You shall know a word by the company it keeps." (Firth 1957)

A bottle of *warzyku* is on the table
Everybody likes *warzyku*
Warzyku makes you drunk
We make *warzyku* out of corn.

- "word meaning" as a feature vector $\vec{w}$ with a binary features f_n
- the words in the context are v_n
- if v_1 is present, the feature f_1 is 1
- here: w = warzyku, v_1 = bottle, v_2 = like, v_3 = drunk, v_4 = corn, v_5 = matrix

Word Similarity

2. Distributional Algorithms

- Intuition: the meaning of a word is related to the distribution of words around it
  - "You shall know a word by the company it keeps." (Firth 1957)

A bottle of *warzyku* is on the table
Everybody likes *warzyku*
Warzyku makes you drunk
We make *warzyku* out of corn.

- "word meaning" as a feature vector $\vec{w}$ with a binary features f_n
- the words in the context are v_n
- if v_1 is present, the feature f_1 is 1
- here: w = warzyku, v_1 = bottle, v_2 = like, v_3 = drunk, v_4 = corn, v_5 = matrix
- word vector: $\vec{w} = (1, 1, 1, 1, 0)$

Word Similarity

- applying distributional algorithms for word similarity measure means deciding about the following facts:
    1. how are the co-occurence terms defined (i.e. what counts as a neighbor)?
    2. how are these terms weighted?
    3. what vector distance metrics are used?

Word Similarity

1. What counts as a neighbor?
   - neighborhoods range from small windows (2 words before and after the target word) to very large context windows (500 words)
   - Schütze (2001)'s experiments show that a context window of 50 words is enough for word sense disambiguation
   - usually, stop words are removed
   - grammatical dependencies and relations can also be used for context vectors

Word Similarity

2. How are the terms weighted?

		relation, w'	*subj-of*, make	*obj-of*, like	*obj-of*, make
target word w	warzyku	f	2	4	1

- vector of N x R features, where R is the number of possible relations
- here: feature f are frequencies (a better indicator than binary values)
- f = (r, w')
- P(f | w) = $\frac{count(f,w)}{count(w)}$ (the probability of feature *f* given a target word *w*)

Word Similarity

		relation, w'	subj-of, make	obj-of, like	obj-of, make
target word w	warzyku	f	2	4	1

- P(f, w) $= \frac{count(f,w)}{\sum_{w'} count(w')}$ (the joint probability of feature *f* given a target word *w* and a context word *w*')

Word Similarity

3. What vector distance metrics are used?

- measure for taking two such vectors and giving a measure of vector similarity
- Levensthein distance: $\text{dist}_L\ (\vec{v},\ \vec{w}) = \sum_{n=1}^{N} |\text{v}_i\ \text{-}\ \text{w}_i|$

Word Similarity

3. What vector distance metrics are used?

- measure for taking two such vectors and giving a measure of vector similarity
- Levensthein distance: $\text{dist}_L\ (\vec{v}, \vec{w}) = \sum_{n=1}^{N} |\text{v}_i \text{ - } \text{w}_i|$
- Euclidean distance: $\text{dist}_E\ (\vec{v}, \vec{w}) = \sqrt{\sum_{n=1}^{N} (v_i - w_i)^2}$

Word Similarity

3. What vector distance metrics are used?

- measure for taking two such vectors and giving a measure of vector similarity
- Levensthein distance: $\text{dist}_L\ (\vec{v},\ \vec{w}) = \sum_{n=1}^{N} |\text{v}_i\ \text{-}\ \text{w}_i|$
- Euclidean distance: $\text{dist}_E\ (\vec{v},\ \vec{w}) = \sqrt{\sum_{n=1}^{N} (v_i - w_i)^2}$
- both measures are rarely used for word similarity, because extreme values change the measure significantly

Word Similarity

- dot product as similarity measure: $\text{dist}_L\ (\vec{v},\ \vec{w}) = \sum_{n=1}^{N} \text{v}_i\ \text{-}\ \text{w}_i$

Word Similarity

- dot product as similarity measure: $\text{dist}_L\ (\vec{v},\ \vec{w}) = \sum_{n=1}^{N} \text{v}_i\ \text{-}\ \text{w}_i$
- BUT: we normalize for the vector length

Word Similarity

- dot product as similarity measure: $\text{dist}_L\ (\vec{v},\ \vec{w}) = \sum_{n=1}^{N} \text{v}_i$ - w_i
- BUT: we normalize for the vector length
- vector length: $|\vec{v}\ | = \sqrt{\sum_{n=1}^{N} v_i^2}$
- normalized dot product:

$$\text{sim}_{norm-dot-product}\ (\vec{v},\ \vec{w}) = \frac{\vec{v} x \vec{w}}{l\vec{v}ll\vec{w}l} = \frac{\sum_{n=1}^{N} v_i - w_i}{\sqrt{\sum_{n=1}^{N} v_i^2}\ \sqrt{\sum_{n=1}^{N} w_i^2}}$$

Semantic Role Labeling

- current approaches rely on on adequate amounts of training and testing data
- General (simplified) approach:
  1. parsing the sentence
  2. finding all predicates (here: verbs)
  3. traversing the tree to determine the roles of the constituents with respect to that predicate

  → feature vector

Semantic Role Labeling

- these observations (feature vectors) are then divided in test and training set
- training of classifier which then yields good results on unlabeled data
- training is mostly done in different stages
  - elimination of some possible role constituents based on simple heuristics (pruning) $\rightarrow$ speeds up training
  - binary identification of each node as being either ARG or NONE
  - classification of the ARG labeled constituents

Towards tracking semantic change by visual analytics

Motivation

1. increasing amount of diachronic data electronically available
2. demand of historical linguists to process these corpora and see developments and patterns over time at-a-glance

Towards tracking semantic change by visual analytics

Motivation

1. increasing amount of diachronic data electronically available
2. demand of historical linguists to process these corpora and see developments and patterns over time at-a-glance

Challenge

Tracking of overall developments of language and also allowing to delve into the details of the data.

Towards tracking semantic change by visual analytics

Motivation

1. increasing amount of diachronic data electronically available
2. demand of historical linguists to process these corpora and see developments and patterns over time at-a-glance

Challenge

Tracking of overall developments of language and also allowing to delve into the details of the data.

Research question

Can we create tools that aid during the analysis of language change, can they test existing hypotheses of change and can they even generate new ones?

Towards tracking semantic change by visual analytics

The object under investigation is semantic change (here: in English)

But what is semantic change?

- if a word changes its meaning over time, it has undergone semantic change.
- some types of semantic change:
  - *narrowing* (the meaning of a word becomes restricted), e.g. skyline
  - *widening* (the meaning of a word widens), e.g. horn
- semantic change in the last 20 years: words related to the computer and the internet

Towards tracking semantic change by visual analytics

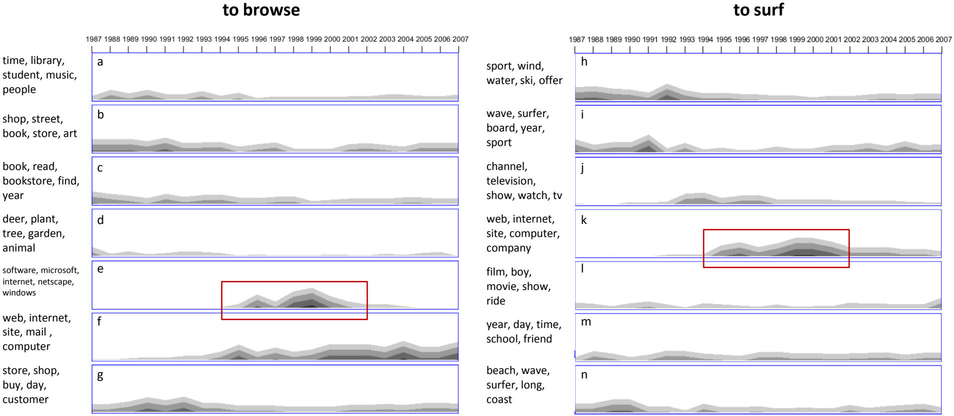
Methodology

- search New York Times corpus
  - 1.8 million newspaper articles from 1987 to 2007
  - each article has a specific time stamp
- extract context of 25 words before and after the lexical item under investigation
- use statistics to model word senses on the basis of word contexts
  - Latent Dirichlet Allocation (LDA) (Blei et al., 2003)
    - not applied on documents but on contexts
  - we predefine the number of senses, each context is assigned to one sense
- add a visualization layer that graphically interprets the information from the statistical analysis and makes it accessible to historical linguists

Towards tracking semantic change by visual analytics

First visualization approach

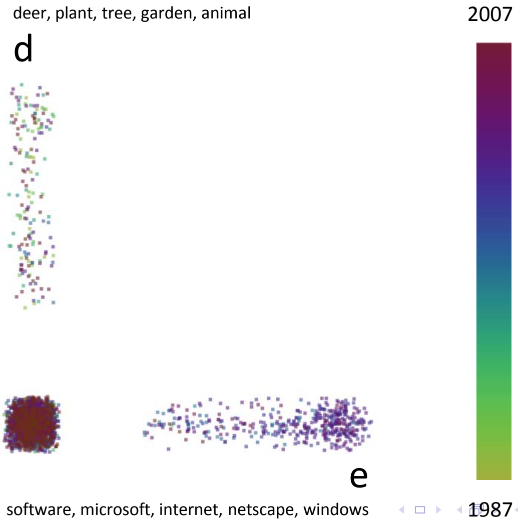
- aggregated view on the data



Towards tracking semantic change by visual analytics

Second visualization approach

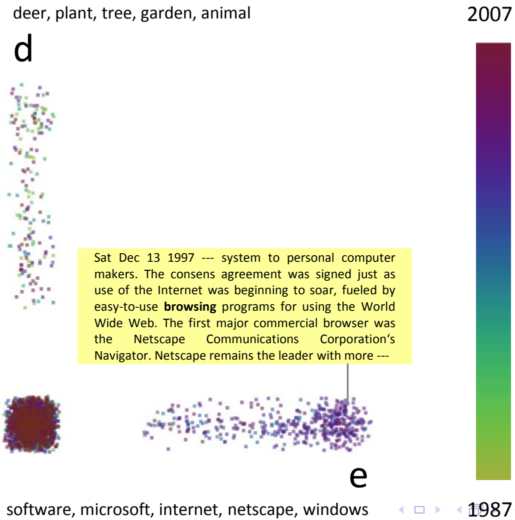
- individual plotting of the contexts of *to browse*



Towards tracking semantic change by visual analytics

Second visualization approach

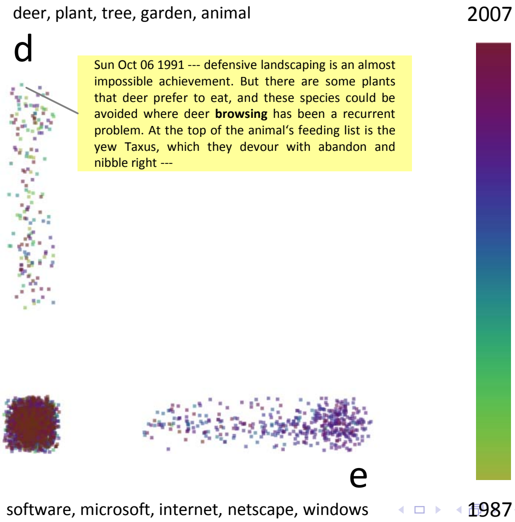
- individual plotting of the contexts of *to browse*



Towards tracking semantic change by visual analytics

Second visualization approach

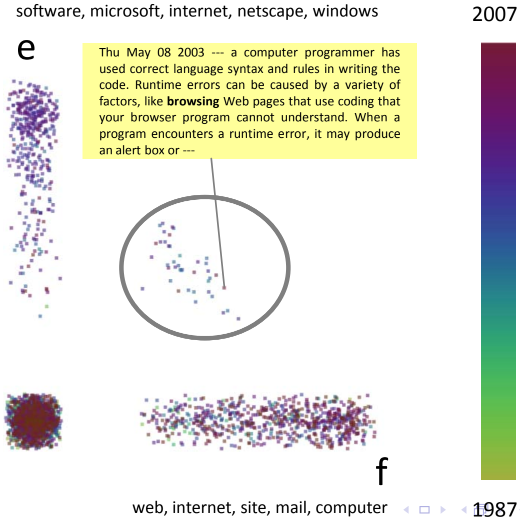
- individual plotting of the contexts of *to browse*



Towards tracking semantic change by visual analytics

Second visualization approach

- individual plotting of the contexts of *to browse*



Towards tracking semantic change by visual analytics

Evaluation

- generally difficult (if not impossible) to fully evaluate statistical approaches to meaning change
- one attempt: compare the findings from the visualization with information from dictionaries from different time periods
  - Longman Dictionary from 1987 (LONG)
  - WordNet from 1998 (WN)
  - Collins dictionary from 2007 (COLL)

Towards tracking semantic change by visual analytics

Evaluation

	to browse		to surf		messenger		bookmark	
	# of word senses		*# of word senses*		*# of word senses*		*# of word senses*	
	DIC	VIS	DIC	VIS	DIC	VIS	DIC	VIS
1987 (LONG)	2	3	1	1	1	2	1	1
1998 (WN)	5	4	3	3	1	3	1	2
2007 (COLL)	3	4	3	2	1	4	2	2

Table: Evaluation of visualized senses against dictionary senses

- in general, the number of our senses corresponds to the information coming from the dictionary
- in the case of "messenger" the visualization proves to be even more detailed

Towards tracking semantic change by visual analytics

Evaluation

	messenger	
	# of word senses	
1987	LONG: a person who brings a message	VIS: bike messenger messenger (genetics)
1997	WN: a person who carries a message	VIS: bike messenger messenger (genetics) religious messenger
2007	COLL: a person who brings a message	VIS: bike messenger messenger (genetics) religious messenger instant messenger

Table: Sense development of *messenger* from 1987 to 2007