NATURAL LANGUAGE PROCESSING

LESSON 7: MORPHOLOGICAL & SEMANTIC AMBIGUITY

OUTLINE

- Morphological Ambiguity & Morphological Disambiguation
  - Surrounding bag of words.
  - POS of neighboring words
  - Local collocations
- Semantic Ambiguity & Word Sense Disambiguation
  - Knowledge-based Approaches
  - Supervised Approaches
  - Unsupervised Approaches
- WordNet Based Disambiguation

Sec. 15.3.1

MORHOLOGICAL AMBIGUITY

- Morphological ambiguity (e.g. lives = live+s or life+s) is a challenging problem for agglutinative languages like Turkish where close to half of the words in running text are morphologically ambiguous.
- A morphological parser for a language with agglutinative morphology, may return more than one possible analysis of a word. This morphological ambiguity needs to be resolved for further language processing.

Sec. 15.3.1

MORHOLOGICAL AMBIGUITY

- Even to decide the part-of-speech tagging of a word, we may need to disambiguate the parses if they have different part-of-speech tags for the final derived word forms.
- For disambiguation of a word, the contextual information for that word is commonly used. We also need a large corpus of sentences to estimate the parameters of the language models in NLP applications.

Sec. 15.3.1

MORHOLOGICAL DISAMBIGUATION

alın+Noun+A3sg+Pnon+Nom (forehead)

al+Adj^DB+Noun+Zero+A3sg+P2sg+Nom (your red)

al+Adj^DB+Noun+Zero+A3sg+Pnon+Gen (of red)

al+Verb+Pos+Imp+A2pl ((you) take)

alın+Verb+Pos+Imp+A2sg ((you) be offended)

Sec. 15.3.1

MORHOLOGICAL DISAMBIGUATION

- As can be seen, some of the parses have different root words and have unrelated morphological features due to the complex morphology of Turkish.
- These ambiguities mostly can be resolved using the contextual information, however the the limited context information cannot resolve the ambiguities.
- Some of the ambiguities can only be solved using semantic or discourse knowledge.

Sec. 15.3.1

MORHOLOGICAL DISAMBIGUATION

- Agglutinative or inflective languages encode more information than just part-of speech tag in a word thanks to the more complex morphology.
- The morphemes that constitute a word carry syntactic and semantic information that is called morphosyntactic and morphosemantic features, respectively.
- For morphological disambiguation, we need to determine all the syntactic morphological features of a word. Therefore morphological disambiguation can be called morphosyntactic tagging in analogy to part-of-speech tagging.

Sec. 15.3.1

MORHOLOGICAL DISAMBIGUATION

Some interpretations of:

Adamı gördüm.

1. I saw the man.
2. I saw my island.

Morphological Ambiguity:

- adam-ı adam+ACC
- ada-m-ı ada+P1SG+ACC

MORHOLOGICAL DISAMBIGUATION

For the noun phrase "*çocuğun kitabı*" (the child's book), the morphological analyzer returns us the following parses:

çocuğun
1. çocuk+Noun+A3sg+Pnon+Gen *(correct parse)*
2. çocuk+Noun+A3sg+P2sg+Nom

kitabı
1. kitap+Noun+A3sg+Pnon+Acc
2. kitap+Noun+A3sg+P3sg+Nom *(correct parse)*

MORHOLOGICAL DISAMBIGUATION

- Assume part-of-speech (POS), e.g. noun, verb, adjective, for the target word is determined.
- Treat as a classification problem with the target word given its POS and morphological features.
- Train a classifier on labeled data encoded using these features.
- Use the trained classifier to disambiguate future instances of the target word given their contextual features.

CONTEXTUAL FEATURES

- Surrounding bag of words.
- POS of neighboring words
- Local collocations

SURROUNDING BAG OF WORDS

- Unordered individual words near the ambiguous word.
- May be words in the same sentence.
- May include words in the previous sentence or surrounding paragraph.

POS OF NEIGHBORING WORDS

- Or differently from the Bag-of-Words approach we can use POS tags of immediately neighboring words.
- Provides evidence of local syntactic context.
- P_{-i} is the POS of the word *i* positions to the left of the target word.
- P_i is the POS of the word *i* positions to the right of the target word.

LOCAL COLLOCATIONS

- Collocation or lexical collocation means two or more words co-occur in a sentence more frequently than by chance.
- A collocation is an expression that forms a specific meaning. It may be noun phrase like *large house*, verbal phrase like *pick up*, idioms or technical terms.
- For example, to determine the word "interest" as a noun, the following collocations are useful:
  - "in the interest of"
  - "an interest in"

SEMANTIC AMBIGUITY

Semantic ambiguity happens when a sentence contains an ambiguous word or phrase that has more than one meaning.

In contrast to morphological ambiguity, in semantic ambiguity the structures of the candidates are the same, but they are interpreted differently.

SEMANTIC AMBIGUITY

Some Turkish Homonym Examples

- Peşine düşen köpekbalığından kaçmak için 3 saat yüzdü!
- Her sabah yüzünü bile yıkamadan kahvesini hazırlar.
- Yüz yılı geçen ömrü ile mahallenin en eski yaşayanıydı.

SEMANTIC AMBIGUITY

Some Turkish Sense Examples:

- Mona Lisa aslında yüzündeki belli belirsiz bir gülümseme ile ilginç bir tablodur.
- Kitabın ön yüzünde biraz yıpranma var, ama durumu iyi.
- İnsanda biraz yüz olur!
- Yastığın yüzünü çıkarıp çamaşır sepetine attı.
- Bu olanlar hep senin yüzünden geliyor başımıza.

SEMANTIC AMBIGUITY

Homonym: each of two or more words having the same spelling but different meanings and origins

yüz (face) – (yüz) hundred | kara (black) – kara (land)

Sense: a way in which an expression or a situation can be interpreted; a meaning.

Turkish has 14,644 words with multiple senses

çıkmak has 56 senses, girmek has 19 senses

gelmek has 36 senses, gitmek has 22 senses

SEMANTIC AMBIGUITY

Words can have more than one distinct meaning:

- 1949: Zipf's «Law of Meaning» More frequent words have more senses than less frequent words -> confirmed later for BNC.
- 1950: Kaplan determined that in a sentence, any ambiguous word's context highly related to two words on both side of the word.
- 1957: Masterman proposed his theory of finding actual sense of a word using the headings of the categories presented in Roget's International Thesaurus

WORD SENSE DISAMBIGUATION (WSD)

WSD is identifying which sense of a word is used in a sentence, when the word has multiple meanings.

The solution to this problem impacts other computer-related writing, such as discourse, improving relevance of search engines, anaphora resolution, coherence, inference, etc.

WSD APPROACHES

- Knowledge-based Approaches
  - Use knowledge sources as machine readable dictionaries, sense inventories or thesauri.
  - **WordNet** (Miller, 1995) is the mostly used
- Supervised Approaches
  - Use machine learning techniques from manually created sense-annotated data.
- Unsupervised Approaches
  - Discriminate the word meanings based on information found in un-annotated corpora.
- Hybrid Approaches

KNOWLEDGE-BASED WSD

The Lesk Algorithm compares the dictionary definition of an ambiguous word with the terms contained in its neighborhood.

Its steps:

1. for every sense of the word being disambiguated one should count the amount of words that are in both neighborhood of that word and in the dictionary definition of that sense
2. the sense that is to be chosen is the sense which has the biggest number of this count

KNOWLEDGE-BASED WSD

The Lesk Algorithm; e.g. “pine cone”.

PINE 1. kinds of evergreen tree with needle-shaped leaves
2. waste away through sorrow or illness

CONE 1. solid body which narrows to a point
2. something of this shape whether solid or hollow
3. fruit of certain evergreen trees

KNOWLEDGE-BASED WSD

Semantic Similarity

Words are related to each other, and share common context. Therefore the smallest semantic distance is chosen as the appropriate sense.

Mostly, a dictionary-thesarus or WordNet is used to find distance between words.

KNOWLEDGE-BASED WSD

Heuristic Method

Heuristics evaluated from different linguistic properties:

- most frequent sense
- one sense per discource
- one sense per collection

Miller et al. 1994: Using A Semantic Concordance for Sense Identification

- Most frequent sense: 58.2% correct
- Random sense: 26.8% correct

SUPERVISED WSD

Decision List Method

The method is a set of «if-then-else» rules induces set of features like feature-value, sense and score are created.

SUPERVISED WSD

Decision List Method

Position	Collocation	beɪs	bæs
Word to the right (+1 w)	bass *player*	105	0
	bass *fishing*	0	94
	bass *are*	0	15
Word to the left (-1 w)	*striped* bass	0	193
	on bass	53	0
	sea bass	0	47
	white bass	0	26
	plays bass	16	0
Within ±20 words (±*k* w)	*fish* (in ±20 words)	0	142
	guitar (in ±20 words)	136	0
	violin (in ±20 words)	49	0
	river (in ±20 words)	0	48
	percussion (in ±20 words)	41	0
	salmon (in ±20 words)	0	38

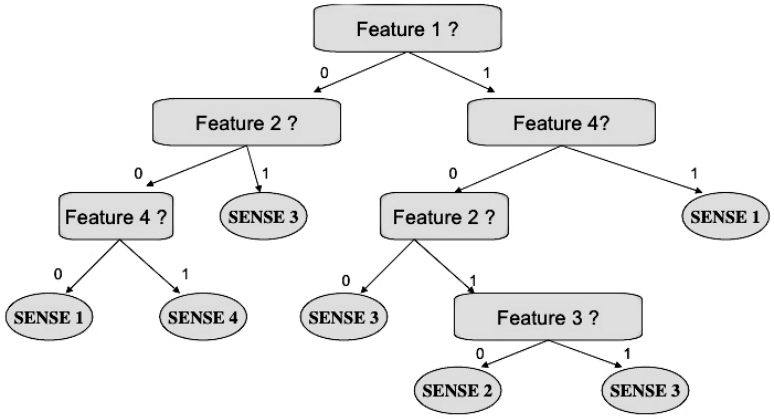
SUPERVISED WSD

Decision Tree Method

It is used to denote classification rules in a tree structure that recursively divides the training set.

SUPERVISED WSD

Decision Tree Method



SUPERVISED WSD

Naive Bayes Method

It is a probabilistic classifier which is based on Bayes Theorem and classifies text documents using two parameters: the conditional probability of each sense (Si) of a word (w) and the features (fj) in the context.

SUPERVISED WSD

Neural Networks Method

Artificial neurons are used for data processing using connectionist approach. Neural Networks can be used to represent words a nodes and these words will activate the ideas to which the are Semantically related.

UNSUPERVISED WSD

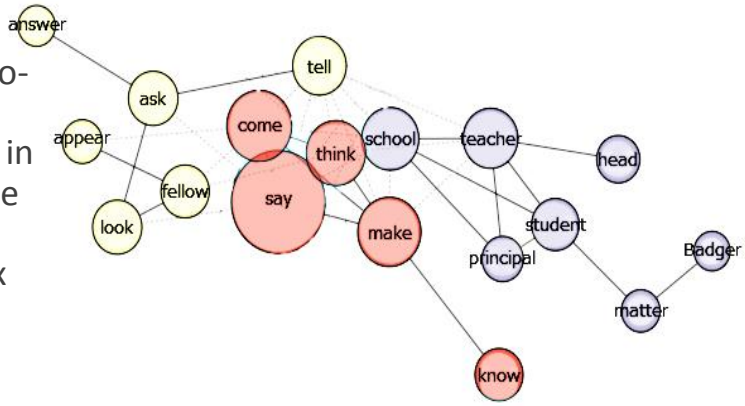
Context Clustering

- Context clustering method is based on clustering techniques in which first context vectors are created and then they will be grouped into clusters to identify the meaning of the word.
- This method uses vector space as word space and its dimensions are words only.
- An a word which is in a corpus will be denoted as vector and how many times it occurs will be counted within its context.
- After that, co-occurence matrix is created and similarity measures applied.
- Then finally discrimination is performed any clustering technique.

UNSUPERVISED WSD

Co-occurrence Graph

This method creates a co-occurrence graph where nodes represent the words in text and edge is added if the words co-occur in the relation according to syntax in the same paragraphs or text.



UNSUPERVISED WSD

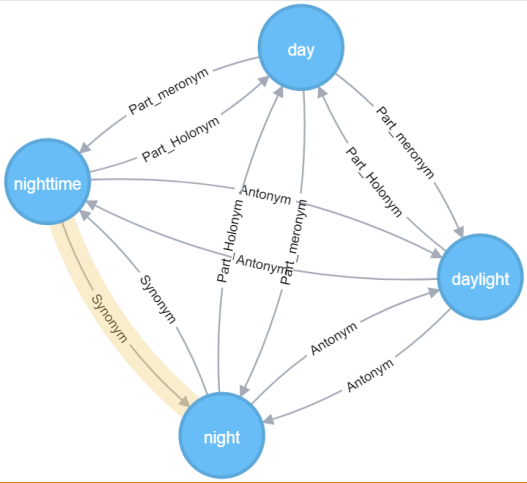
Co-occurrence Graph

- For a given word, first, the graph is created and the adjacency matrix for the graph created.
- Markov clustering method is applied to find the meaning of the word.
- Each edge of graph is assigned a weight which is co-occurring frequency of those words.
- Word with high frequency is assigned 0 and words with rarely co-occurring assigned 1.

WordNet

- WordNet is an electronic lexical database developed at Princeton University.
- Development has started in 1985, and still goes on.
- Currently version 3.1 and older versions are publicly available from https://wordnet.princeton.edu/wordnet/
- WordNet entries (“senses”) are organized into synonyms sets (“synsets”) representing concepts.
- Each synset in WordNet is followed by its definition (“gloss”).
- WordNet supports semantic relations, which link concepts (i.e. synsets), such as hypernymy, hyponymy, meronymy, holonymy, troponymy etc.

WordNet



WordNet

- WordNet includes the lexical categories nouns, verbs, adjectives and adverbs but ignores prepositions, determiners and other function words.
- Words from the same lexical category that are roughly synonymous are grouped into synsets.
- The different senses of a polysemous word form are assigned to different synsets.
- The meaning of a synset is further clarified with a short defining gloss and one or more usage examples

WordNet

- WordNet 3.0 Statistics

POS	Unique Strings	Synsets	Total Word-Sense Pairs
Noun	117,798	82,115	146,312
Verb	11,529	13,767	25,047
Adjective	21,479	18,156	30,002
Adverb	4,481	3,621	5,580
Totals	155,287	117,659	206,941

WordNet

- WordNet 3.0 Statistics

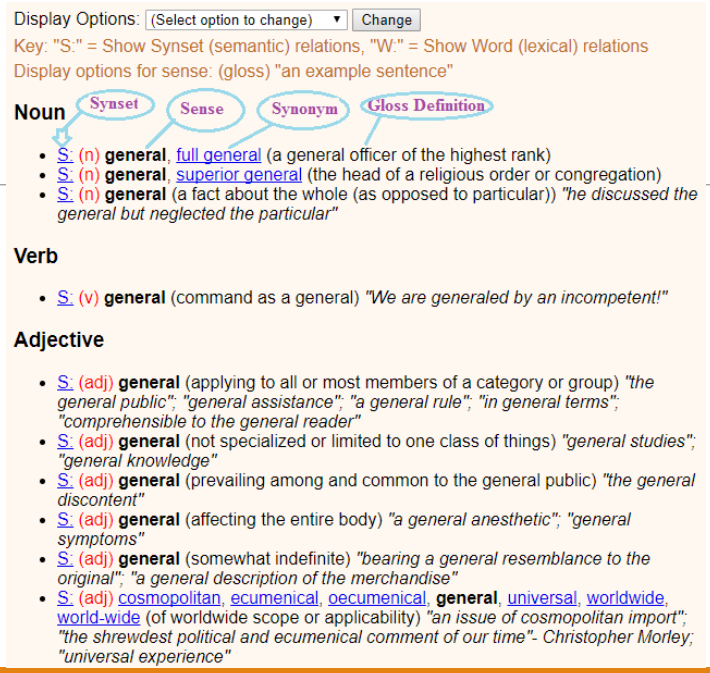
POS	Monosemous Words and Senses	Polysemous Words	Polysemous Senses
Noun	101,863	15,935	44,449
Verb	6,277	5,252	18,770
Adjective	16,503	4,976	14,399
Adverb	3,748	733	1,832
Totals	128,391	26,896	79,450

WordNet

Online version of WordNet

http://wordnetweb.princeton.edu/perl/webwn

WordNet

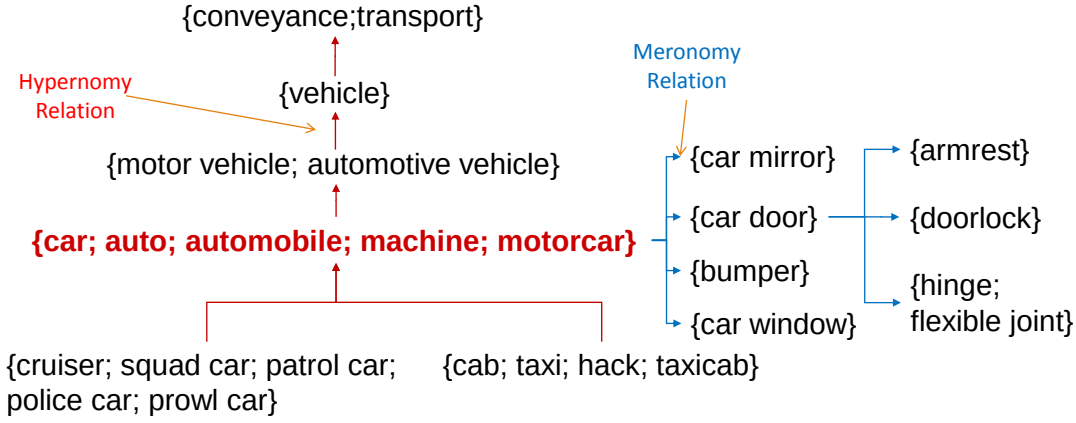


WordNet

- WordNet desktop version is available to download and use on a PC
- WordNet DB is also distributed publicly in internet.
- Data and index files are provided in simple text format.
- File format and expressions are also documented and shared.
- Data and index files for nouns, adjectives, verbs and adverbs are provided.

WordNet

Semantic Structure and Relations in WordNet



BalkaNet - TURKISH WordNet

- BalkaNet was a project funded by EU during 2001-2004, purpose of the project was to develop a WordNet for the six Balkan Countries Languages (Turkish, Rumanian, Bulgarian, Greek, Serbian and Czech).
- It has the same structure with WordNet 2.0 version.
- Same structure is used for the synsets and relations there are fewer relation defined in BalkaNet.
- It is currently eleven relation defined.
- BalkaNet is available as an XML file to the researchers.

WordNet BASED WSD METHODS

- Main aim to develop a WordNet is to find the «relatedness» between two concepts.
- If there is an ambigous word, it means there are several synsets which are represented by the same words.
- Generally synsets with multiple senses are disambiguated using relatedness metrics in WordNet.
- Relatedness is measured using the other words in the context.
- So context is very important for WSD, for a word alone only probablistic methods can estimate the correct sense.

WordNet BASED WSD METHODS

- If a amibugus word is given in n-gram, then all the senses of the words and target ambigus words are compared in terms of semantic relatedness.
- Highest relatedness between those senses are considered as the correct senses.
- Example: *manager fired worker*
- First all the tokens are lemmatized, pos tagged and then below word list is found
- w_1=manager$_n$ w_2=fire$_v$ w_3=worker$_n$

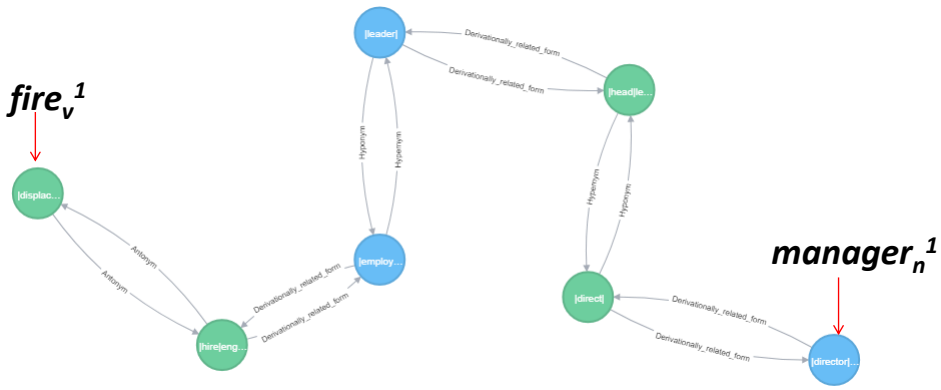
WordNet BASED WSD METHODS

$$\text{argmax}\left(\begin{bmatrix} manager_n^1 \\ manager_n^2 \\ . \\ . \\ . \\ manager_n^m \end{bmatrix} X \begin{bmatrix} fire_v^1 \\ fire_v^2 \\ . \\ . \\ . \\ fire_v^t \end{bmatrix} X \begin{bmatrix} worker_n^1 \\ worker_n^2 \\ . \\ . \\ . \\ worker_n^k \end{bmatrix}\right)$$

As you see all the words in the list have multiple senses, so each senses of the each word is compared with each senses of other words by means of relatedness

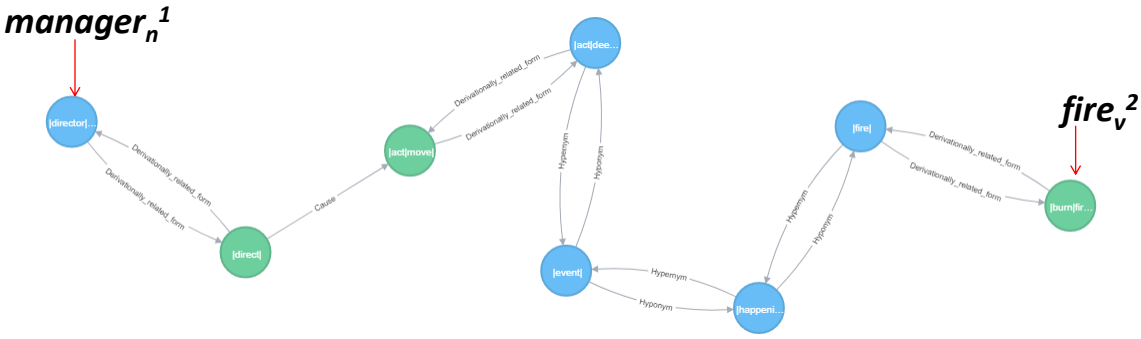
WordNet BASED WSD METHODS

Shortest Path in WordNet between ***manager$_n^1$ and fire$_v^1$***



WordNet BASED WSD METHODS

Shortest Path in WordNet between ***manager$_n^1$ and fire$_v^2$***



WordNet BASED WSD METHODS

- And maximum reletedness between senses of the two words (source word and destination word) are accounted as the correct sense of source word.
- Max(rel(w_1s_n,w_as_b)) where w_1s_n= w_1s_1, w_1s_2,… w_1s_n

w_as_b= w_2s_1, w_2s_2,… w_2s_b

. . .

. . .

w_as_1, w_as_2,… w_as_b

WordNet BASED WSD METHODS

- To measure the relation between the senses, there several methods are used, these methods uses some metric.
- To find the relation, WordNet is considered as a graph or tree structure.
- Path based, depth based, both used for the measurement.
- Also in the recent years, probablity distribution vectors of each senses are used for the relatedness.
- In the next chapters these methods will be described in details