NATURAL LANGUAGE PROCESSING

LESSON 7: AMBIGUITY

1

OUTLINE

- Ambiguity
  - Lexical
  - Syntactic
  - Morphological
  - Semantic
- Word Sense Disambiguation
  - Knowledge-based Approaches
    - LESK vs WordNet
  - Supervised & Unsupervised Approaches

2

AMBIGUITY

Ambiguity is originally used for topics that can confuse a person when trying to understand a natural language. But now that we teach machines to process natural languages as well, we use the term broadly to represent all the dilemmas in analysis.

- Lexical Ambiguity
- Syntactic Ambiguity
- Morphological Ambiguity
- Semantic Ambiguity

3

Sec. 15.3.1

LEXICAL AMBIGUITY

- The term lexical ambiguity, which is used for words with spelling similarity, is often explained within the scope of semantic ambiguity.
- For instance, the word "bank" has several distinct lexical definitions, including "financial institution" and "edge of a river".
- Or consider "apothecary". One could say "I bought herbs from the apothecary". This could mean one actually spoke to the apothecary (pharmacist) or went to the apothecary (pharmacy).

4

SYNTACTIC AMBIGUITY

- This form of ambiguity is also called structural or grammatical ambiguity. It occurs in the sentence because the sentence structure leads to two or more possible meanings.
- For example, the sentence 'Papa ate the caviar with a spoon' has a syntactic ambiguity. This is why you can find at least two syntactic parsing.
- Although we call this problem syntactic ambiguity, it also affects the meaning of the text. Therefore, semantic analysis may be required in its solution.

5

MORPHOLOGICAL AMBIGUITY

- Morphological ambiguity is a challenging problem for agglutinative languages like Turkish where close to half of the words in running text are morphologically ambiguous.
- In English, we can give an example as 'lives'. We can parse it as live + s or life + s.
- 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.

6

MORPHOLOGICAL AMBIGUITY

Let’s look at the Turkish word ‘ALIN’

- 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 (take)
- alın+Verb+Pos+Imp+A2sg (be offended)

7

MORPHOLOGICAL AMBIGUITY

Some interpretations of:

Adamı gördüm.

Morphological Ambiguity:

◦ adam-ı	adam+ACC	(I saw the man)
◦ ada-m-ı	ada+P1SG+ACC	(I saw my island)

8

MORPHOLOGICAL DISAMBIGUATION

Morphological disambiguation is the task of determining the contextually correct morphological parses of tokens in a sentence.

A morphological disambiguator takes in sets of morphological parses for each token, generated by a morphological analyzer, and then selects a morphological parse for each, considering linguistic and contextual information by a statistical or a neural method.

9

SEMANTIC AMBIGUITY

Semantic ambiguity happens when a sentence contains an ambiguous word or phrase that has more than one meaning. In semantic ambiguity the structures of the candidates are the same, but they are interpreted differently.

Here, we have 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.

10

SEMANTIC AMBIGUITY

Although homonym is confused, semantic ambiguity focuses precisely on sense ambiguity. Below are some Turkish 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.

11

WORD SENSE DISAMBIGUATION (WSD)

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

In humans, it is usually done quickly by subconscious. But in natural language processing, it is still serious one of the big problems.

Many techniques have been researched, including knowledge-based methods, supervised machine learning methods, and completely unsupervised methods.

12

WSD APPROACHES

- Knowledge-based Approaches
  - Lesk (Lesk, 1986).
  - **WordNet** (Miller, 1995)
- Supervised Approaches
  - Machine learning techniques on sense-annotated data.
- Unsupervised Approaches
  - Cluster word meanings using un-annotated corpora.

13

KNOWLEDGE-BASED WSD

The Lesk Algorithm is a classical algorithm for word sense disambiguation introduced by Michael E. Lesk in 1986. It is based on the assumption that words in a given "neighborhood of text" will tend to share a common topic. Its steps:

1. The dictionary definition of each sense is found. Then, the dictionary definitions of the ambiguous word's neighbors in the target text are also determined.
2. The senses of the ambiguous word and their neighbors are compared in terms of dictionary definitions and the sense with the most common word with its neighbors is selected.

14

KNOWLEDGE-BASED WSD

The most common example to show that the Lesk algorithm works is the phrase "pinecone".

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

15

KNOWLEDGE-BASED WSD

But the Lesk algorithm has a serious handicap in expecting the words in the dictionary to match literally.

This method was later extended using WordNet.

We will discuss this in detail shortly.

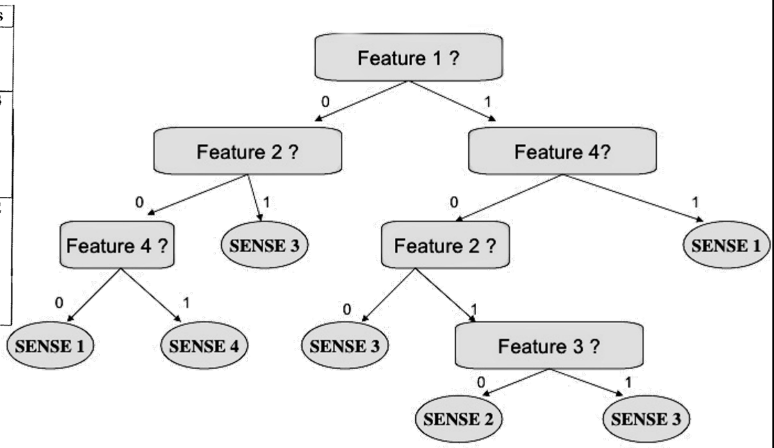
16

SUPERVISED WSD

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

Decision List

Decision Tree



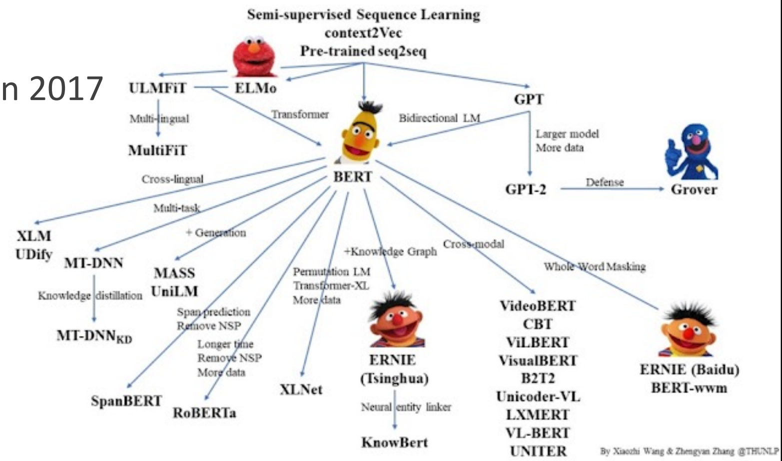
17

SUPERVISED WSD

Neural Networks

Transformers by Google in 2017

BERT by Google in 2018



18

UNSUPERVISED WSD

Unsupervised methods that do word-sense disambiguation basically have two approaches:

1. Using a network whose inter-word relationships are defined by experts, such as WordNet.
2. Using inter-word co-occurrence relations with the help of a large corpus.

19

UNSUPERVISED WSD

Whether WordNet or Corpus is used as dataset, the next step is the calculation method. For this stage, we can talk about two different approaches:

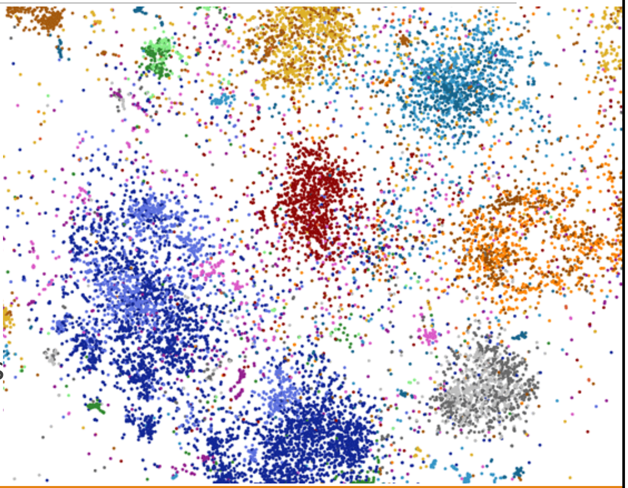
1. to represent words as a vector in space and uses space geometry in calculations.
2. to represent words in a graph network and graph algorithms are used for calculations in these approaches.

20

UNSUPERVISED WSD

Context cluster method

In this approach, if the word groups are observed as separate clusters, as in the figure, it becomes easier to interpret the sense of any ambiguous word. It is thought that the ambiguous word is used in the sense closest to that cluster, in whichever cluster the majority of its neighbors are found.

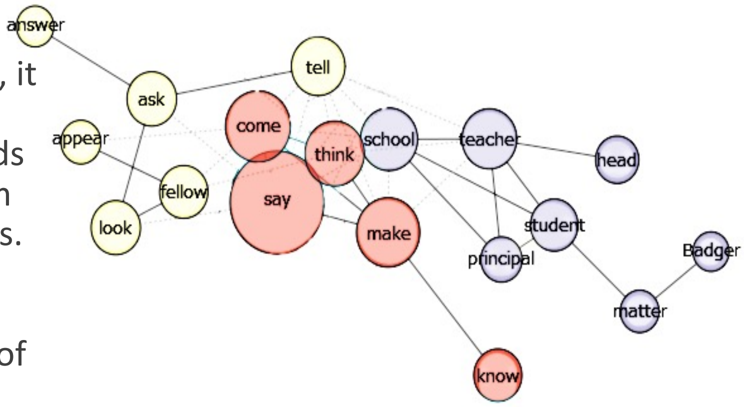


21

UNSUPERVISED WSD

Graph method

After the graph is prepared, it is possible to calculate the distance between two words as the number of hops from the nodes using some filters. Thus, the ambiguity is removed by choosing the closest sense to the nodes of the neighboring words.



22

WordNet

- WordNet is an electronic lexical database developed at Princeton University.
- Development has started in 1985, and still goes on.
- Last and older versions are publicly available on its web site wordnet.princeton.edu.
- WordNet entries are organized into SYNonyms SETs ("synset") representing senses.
- WordNet supports semantic relations, which link concepts (i.e. synsets), such as hypernymy, hyponymy, meronymy, holonymy, troponymy etc.

23

WordNet

If synonyms had not been put in a single node with the name "synset", the number of relationships would have increased tremendously. For example in the figure, although the words Nighttime and Night are synonymous, they are designed as two separate nodes. Because of this design, their antonym to the Daylight node had to be defined twice.

24

WordNet

Some Statistics from WordNet 3.0

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

25

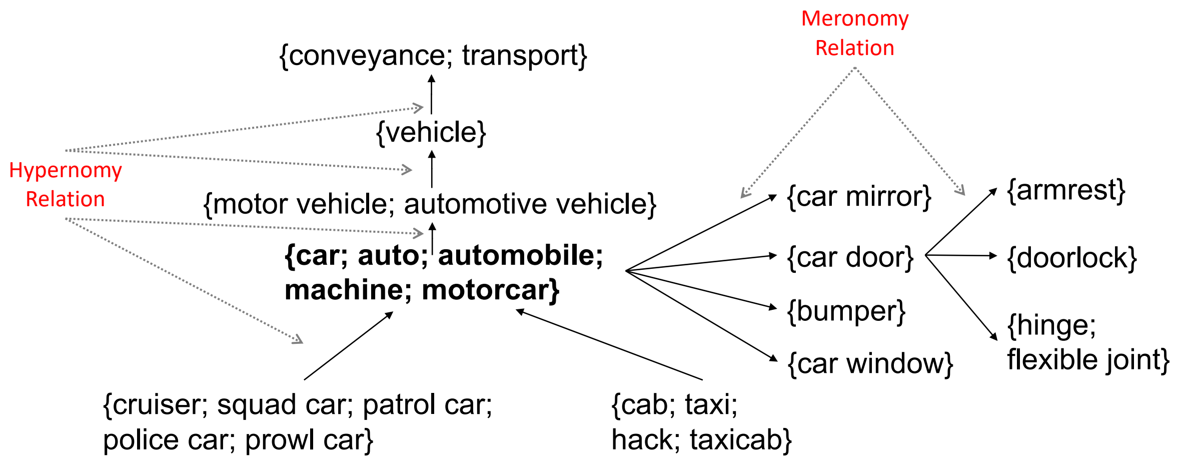
WordNet

Some statistical information is presented in the table. It seems clear that the number of words in the noun type alone represents three-quarters of the entire WordNet.

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

26

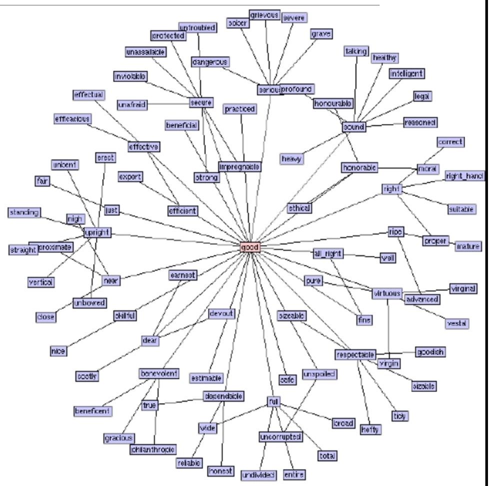
WordNet



27

WordNet based WSD METHODS

- There are many methods in the literature and the calculation approach is different in each.
- The distance between two nodes is calculated according to the approach in the method used.
- Here, the hop number will be used regardless of the relationship type for easy understanding.



28

WordNet based WSD METHODS

- In the example below, both the words "manager" and "fired" have multiple senses. Let's try to find the most suitable senses.

  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=$worker_n$

29

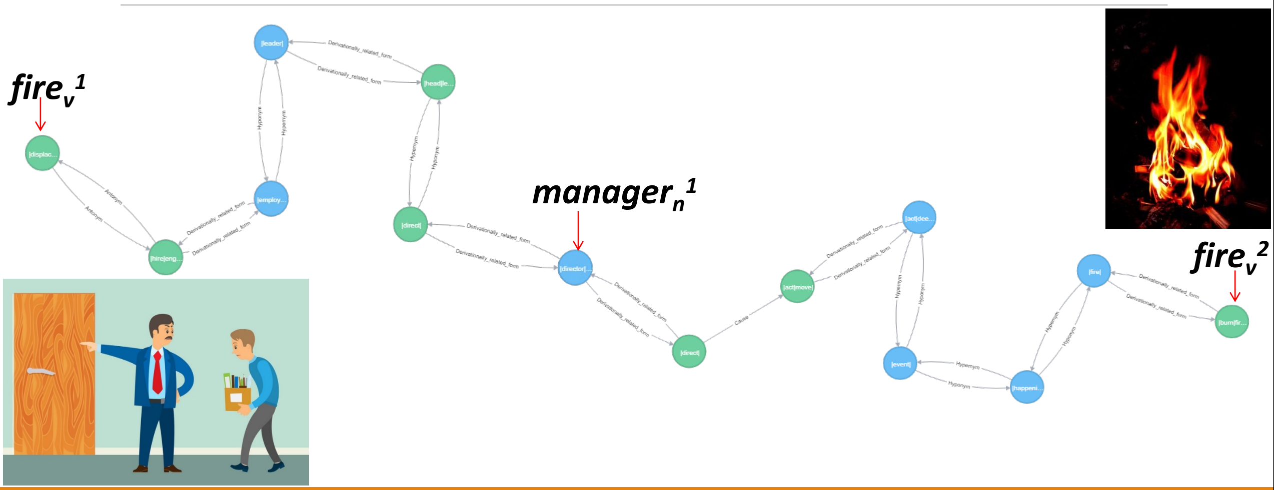
WordNet based WSD METHODS

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

As you can see, all the words in the list have more than one sense. Inspired by the Viterbi algorithm, the senses of each word are compared by being related to the senses of other words.

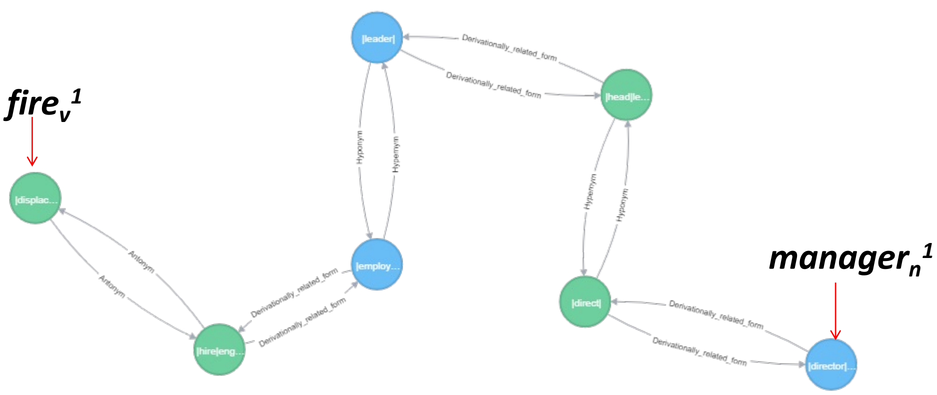
30

WordNet based WSD METHODS



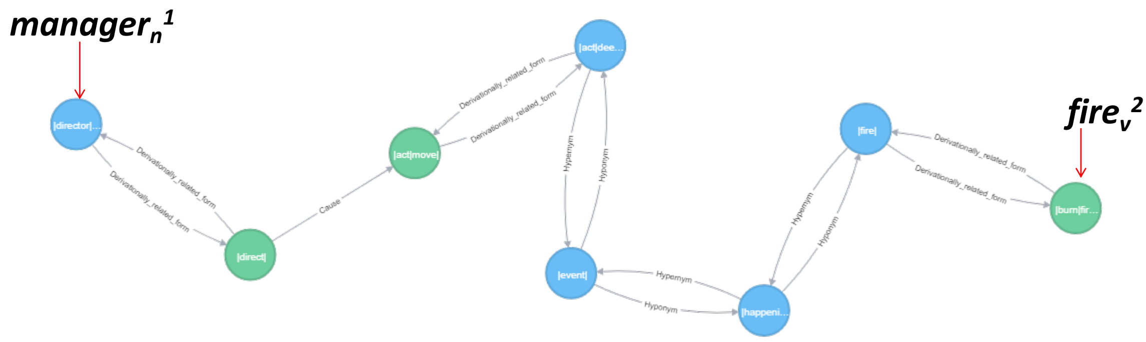
31

WordNet based WSD METHODS



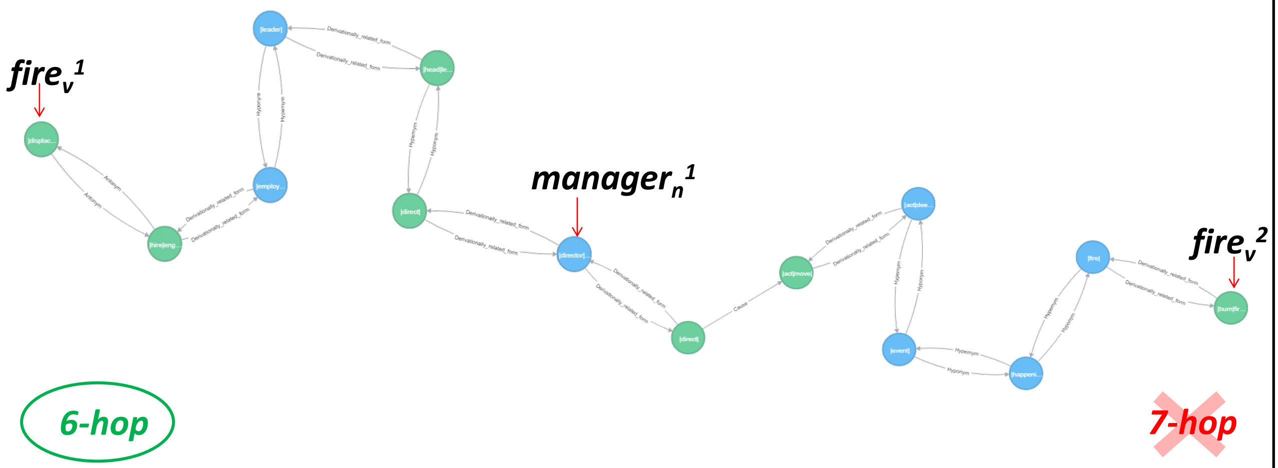
32

WordNet based WSD METHODS



33

WordNet based WSD METHODS



34