Natural Language Processing

Assist. Prof. Dr. Tuğba YILDIZ

İSTANBUL BİLGİ UNIVERSITY
Department of Computer Engineering

May 5, 2017

Lexical Semantics

- Traditionally, meaning in language has been studied from three perspectives:
- The meanings of individual words
- How those meanings combine to make meanings for individual sentences or utterances
- How those meanings combine to make meanings for a text or discourse
- We are going to focus today on word meaning, also called lexical semantics

Preliminaries

- Lexeme: An entry in a lexicon consisting of a pairing of a form with a single meaning representation
- Lexicon: A collection of lexemes

Semantic Relations

- Semantic is about meaning of word or phrases
- Semantic relations are underlying relations between two concepts expressed by words or phrases
- Play an essential role in lexical semantics applications:
- Question-Answering
- Text-Summarization
- Lexico Semantic Knowledge Bases, etc.

Relationships between word meanings

- Polysemy
- Synonymy
- Antonymy
- Hypernomy/Hyponomy
- Meronomy/Holonomy

Polysemy

- The bank is constructed from red brick
- I withdrew the money from the bank
- Are those the same sense?
- A single lexeme with multiple related meanings (bank the building, bank the finantial institution)
- Most non-rare words have multiple meanings
- The number of meanings is related to its frequency
- Verbs tend more to polysemy

Synonym

- Word that have the same meaning in some or all contexts.
  - automobile-car
  - big-large
  - youth-adolescent
- Two lexemes are synonyms if they can be successfully substituted for each other in all situations

Synonym

- But there are few (or no) examples of perfect synonymy.
- Even if many aspects of meaning are identical
- Example:
  Big and large?
  That's my big sister
  That's my large sister

Antonym

- Words that are opposites with respect to one feature of their meaning
- Otherwise, they are very similar!
  - dark-light
  - hot-cold
  - boy-girl !

Antonym

- Words that are opposites with respect to one feature of their meaning
- Otherwise, they are very similar!
  - dark-light
  - hot-cold
  - boy-girl !

Word Similarity Computation

- For various computational applications it's useful to find words which are similar to another word.
- Machine translation (to find near-synonyms)
- Information retrieval (to do "query expansion")
- Two ways to do this:
  - Automatic computation based on distributional similarity
  - Use a thesaurus which lists similar words.
    WordNet

Hyponymy

- Hyponymy: the meaning of one lexeme is a subset of the meaning of another
- Since dogs are canids
- Dog is a hyponym of canid and
- Canid is a hypernym of dog
- Similarly,
- Car is a hyponym of vehicle
- Vehicle is a hypernym of car

Meronymy

- a non-hierarchical relationship between terms that respect to the significant parts of a whole.
- "the eye is part of the face"
  eye: part
  face: whole
- eye is a meronym of face
- face is a holonym of eye

Semantic Lexicons

- Important for NLP tasks, especially building semantic lexicons.
- Solutions:
  WordNet (Miller 1990)
  CYC (Lenat, Prakash, and Shepherd 1986)
- Benefits:
  reliable, effective and widely used
- Problems: troublesome, time-consuming and cost of extention and maintenance operation

Semantic Lexicons

- Solutions:
- Automatically extract knowledge from some sources
- Sources for Discovery of Semantic Relation:
  Web
  Documents
  News
  Corpus
  Dictionary

WordNet

- A hierarchically organized lexical database

Category	Unique Forms	Number of Senses
Noun	94474	116317
Verb	10319	22066
Adjective	20170	29881
Adverb	4546	5677

Fig.1 Scope of the current WordNet 1.6 release in terms of unique entries and total number of senses for the four databases.

WordNet

- A hierarchically organized lexical database

The noun "bass" has 8 senses in WordNet.
1. bass - (the lowest part of the musical range)
2. bass, bass part - (the lowest part in polyphonic music)
3. bass, basso - (an adult male singer with the lowest voice)
4. sea bass, bass - (flesh of lean-fleshed saltwater fish of the family Serranidae)
5. freshwater bass, bass - (any of various North American lean-fleshed freshwater fishes especially of the genus Micropterus)
6. bass, bass voice, basso - (the lowest adult male singing voice)
7. bass - (the member with the lowest range of a family of musical instruments)
8. bass - (nontechnical name for any of numerous edible marine and freshwater spiny-finned fishes)

Fig.2 The WordNet 1.6 entry for the noun bass.

WordNet

- Noun Relations

Relation	Definition	Example
Hyperym	From concepts to superordinates	*breakfast* → *meal*
Hyponym	From concepts to subtypes	*meal* → *lunch*
Has-Member	From groups to their members.	*faculty* → *professor*
Member-Of	From members to their groups.	*copilot* → *crew*
Has-Stuff	From things to what they're made of.	→
Stuff-Of	From stuff to what it makes up.	→
Has-Part	From wholes to parts	*table* → *leg*
Part-Of	From parts to wholes.	*course* → *meal*
Antonym	Opposites	*leader* → *follower*

Fig.3 Noun Relations in WordNet

WordNet

- Verb Relations

Relation	Definition	Example
Hypernym	From events to superordinate events	*fly* → *travel*
Troponym	From events to their subtypes	*walk* → *stroll*
Entails	From events to the events they entail	*snore* → *sleep*
Antonym	Opposites	*increase* ⟺ *decrease*

Fig.4 Verb Relations in WordNet

WordNet

- Adjective and Adverb Relations

Relation	Definition	Example
Antonym	Opposite	*heavy* $\Longleftrightarrow$ *light*
Adverb	Opposite	*quickly* $\Longleftrightarrow$ *slowly*

Fig.5 Adjective and Adverb Relations in WordNet

WordNet

- Hyponym chains:

```
Sense 3
bass, basso --
(an adult male singer with the lowest voice)
=> singer, vocalist
   => musician, instrumentalist, player
      => performer, performing artist
         => entertainer
            => person, individual, someone...
               => life form, organism, being...
                  => entity, something
               => causal agent, cause, causal agency
                  => entity, something

Sense 7
bass --
(the member with the lowest range of a family of
musical instruments)
=> musical instrument
   => instrument
      => device
         => instrumentality, instrumentation
            => artifact, artefact
               => object, physical object
                  => entity, something
```

Fig.6 Hyponymy chains for two separate senses of the lexeme bass.

WordNet

- wn auto
- wn auto -meron
- wn auto -synsn
- wn auto -over
- wn dog -hypev

WordNet

- NLTK

```
>>>from nltk.corpus import wordnet as wn
>>>wn.synsets('dog')
[Synset('dog.n.01'), Synset('frump.n.01'), Synset('dog.n.03'),
Synset('cad.n.01'), Synset('frank.n.02'), Synset('pawl.n.01'),
Synset('andiron.n.01'), Synset('chase.v.01')]
>>>dog = wn.synset('dog.n.01')
>>>dog.hypernyms()
[Synset('canine.n.02'), Synset('domestic_animal.n.01')]
>>>dog.hyponyms() [Synset('basenji.n.01'),
Synset('corgi.n.01') , Synset('cur.n.01'),
Synset('dalmatian.n.02'), ...]
>>>dog.member_holonyms()
[Synset('canis.n.01'), Synset ('pack.n.06')]
>>>dog.root_hypernyms()
[Synset('entity.n.01')]
```

WordNet

- Synonymy in WordNet is organized around the notion of a synset, a set of synonyms.
- Consider the following example of a synset
  {chump, fish, fool, gull, mark, patsy, fall guy,sucker, schlemiel, shlemiel, soft touch, mug}

WordNet

- The critical thing to grasp about WordNet is the notion of a synset; it's their version of a sense or a concept
- Example: table as a verb to mean defer
  postpone, hold over, table, shelve, set back, defer, remit, put off
- For WordNet, the meaning of this sense of table is this list.
- Another example: give has 45 senses in WN
  supply, provide, render, furnish,....

Word Senses

- The meaning of a word in a given context
- Word sense representations
- With respect to a dictionary
  chair = a seat for one person, with a support for the back; "he put his coat over the back of the chair and sat down"
  chair = the position of professor; "he was awarded an endowed chair in economics"
- With respect to the translation in a second language
  chair = chaise
  chair = directeur
- With respect to the context where it occurs (discrimination)
  "Sit on a chair" "Take a seat on this chair"
  "The chair of the Math Department" "The chair of the meeting"

Word Sense Disambiguation (WSD)

- Given a word in context, decide which sense of the word this is.

Approaches to Word Sense Disambiguation(WSD)

- Knowledge-Based Disambiguation
  - use of external lexical resources such as dictionaries and thesauri
  - discourse properties
- Supervised Disambiguation
  - based on a labeled training set
  - the learning system has:
    - a training set of feature-encoded inputs AND
    - their appropriate sense label (category)
- Unsupervised Disambiguation
  - based on unlabeled corpora
  - The learning system has:
    - a training set of feature-encoded inputs BUT
    - NOT their appropriate sense label (category)

Knowledge-based WSD

- class of WSD methods relying (mainly) on knowledge drawn from dictionaries and/or raw text
- Resources:
- YES
  - Machine Readable Dictionaries
  - Raw corpora
- NO
  - Manually annotated corpora
  - Scope
- All open-class words

Knowledge-based WSD

- In recent years, most dictionaries made available in Machine Readable format (MRD)
  - Oxford English Dictionary
  - Collins
  - Longman Dictionary of Ordinary Contemporary English (LDOCE)
- Thesauruses - add synonymy information
  - Roget Thesaurus
- Semantic networks - add more semantic relations
  - WordNet
  - EuroWordNet

Knowledge-based WSD

- MRD - A Resource for Knowledge-based WSD
  - A thesaurus adds: An explicit synonymy relation between word meanings

    WordNet synsets for the noun "plant"
    1. plant, works, industrial plant
    2. plant, flora, plant life
  - A semantic network adds: Hypernymy/hyponymy (IS-A), meronymy/holonymy (PART-OF), antonymy, entailnment, etc.

    WordNet related concepts for the meaning "plant life" {plant, flora,
    hypernym: {organism, being}
    hypomym: {house plant}, {fungus},
    meronym: {plant tissue}, {plant part}
    holonym: {Plantae, kingdom Plantae, plant kingdom}

Knowledge-based WSD: Algorithms based on Machine Readable Dictionaries

- Lesk Algorithm (Michael Lesk 1986): Identify senses of words in context using definition overlap
  1. Retrieve from MRD all sense definitions of the words to be disambiguated
  2. Determine the definition overlap for all possible sense combinations
  3. Choose senses that lead to highest overlap

Knowledge-based WSD: Algorithms based on Machine Readable Dictionaries

- Lesk Algorithm (Michael Lesk 1986): Identify senses of words in context using definition overlap
- Example: disambiguate 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

Pine#1 ∩ Cone#1 = 0
Pine#2 ∩ Cone#1 = 0
Pine#1 ∩ Cone#2 = 1
Pine#2 ∩ Cone#2 = 0
Pine#1 ∩ Cone#3 = 2
Pine#2 ∩ Cone#3 = 0

Knowledge-based WSD: Algorithms based on Machine Readable Dictionaries

- Lesk Algorithm for More than Two Words?
- I saw a man who is 98 years old and can still walk and tell jokes
- nine open class words: see(26), man(11), year(4), old(8), can(5), still(4), walk(10), tell(8), joke(3)
- 43,929,600 sense combinations! How to find the optimal sense combination?

Knowledge-based WSD: Algorithms based on Machine Readable Dictionaries

- Lesk Algorithm: A Simplified Version
- Original Lesk definition: measure overlap between sense definitions for all words in context
  - Identify simultaneously the correct senses for all words in context
- Simplified Lesk (Kilgarriff & Rosensweig 2000): measure overlap between sense definitions of a word and current context
  - Identify the correct sense for one word at a time
  - Search space significantly reduced

Knowledge-based WSD: Algorithms based on Machine Readable Dictionaries

- Lesk Algorithm: A Simplified Version
  1. Retrieve from MRD all sense definitions of the word to be disambiguated
  2. Determine the overlap between each sense definition and the current context
  3. Choose the sense that leads to highest overlap
- Example: disambiguate PINE in "Pine cones hanging in a tree"
- PINE :
  1. kinds of evergreen tree with needle-shaped leaves
  2. waste away through sorrow or illness

Pine#1 ∩ Sentence = 1
Pine#2 ∩ Sentence = 0

Knowledge-based WSD: Algorithms based on Machine Readable Dictionaries

- Evaluations of Lesk Algorithm
- Evaluation on Senseval-2 all-words data, with back-off to random sense (Mihalcea & Tarau 2004)
  - Original Lesk: 35%
  - Simplified Lesk: 47%
- Evaluation on Senseval-2 all-words data, with back-off to most frequent sense (Vasilescu, Langlais, Lapalme 2004)
  - Original Lesk: 42%
  - Simplified Lesk: 58%

Knowledge-based WSD

- Selectional Preferences : A way to constrain the possible meanings of words in a given context
- Example:
  - E.g. "Wash a dish" vs. "Cook a dish"
  - WASH-OBJECT vs. COOK-FOOD
- Capture information about possible relations between semantic classes
- Common sense knowledge
- Alternative terminology:
  - Selectional Restrictions
  - Selectional Preferences
  - Selectional Constraints

Knowledge-based WSD

- From annotated corpora
- Circular relationship with the WSD problem
- Example:
  - Need WSD to build the annotated corpus
  - Need selectional preferences to derive WSD
- From raw corpora
  - Frequency counts
  - Information theory measures
  - Class-to-class relations

Knowledge-based WSD

- Preliminaries: Learning Word-to-Word Relations
- An indication of the semantic fit between two words
- 1. Frequency counts
  Pairs of words connected by a syntactic relations
  Counts(W1,W2,Rel)
- 2. Conditional probabilities
- Condition on one of the words

$$P(W_1|W_2,R)=\frac{Count\ (W_1,W_2,R)}{Count\ (W_2,R)}$$

Knowledge-based WSD

- Learning Selectional Preferences
- Determine the contribution of a word sense based on the assumption of equal sense distributions:
- e.g. "plant" has two senses 50% occurrences are sense 1, 50% are sense 2

Example: learning restrictions for the verb "to drink"

– Find high-scoring verb-object pairs

Co-occ score	Verb	Object
11.75	drink	tea
11.75	drink	Pepsi
11.75	drink	champagne
10.53	drink	liquid
10.2	drink	beer
9.34	drink	wine

– Find "prototypical" object classes (high association score)

A(v,c)	Object class
3.58	(beverage, [drink, …])
2.05	(alcoholic_beverage, [intoxicant, …])

Knowledge-based WSD

- Learning Selectional Preferences

Algorithm:

1. Learn a large set of selectional preferences for a given syntactic relation R
2. Given a pair of words W_1– W_2 connected by a relation R
3. Find all selectional preferences W_1– C (word-to-class) or C_1– C_2 (class-to-class) that apply
4. Select the meanings of W_1 and W_2 based on the selected semantic class

Example: disambiguate *coffee* in "drink *coffee*"

1. (beverage) a beverage consisting of an infusion of ground coffee beans
2. (tree) any of several small trees native to the tropical Old World
3. (color) a medium to dark brown color

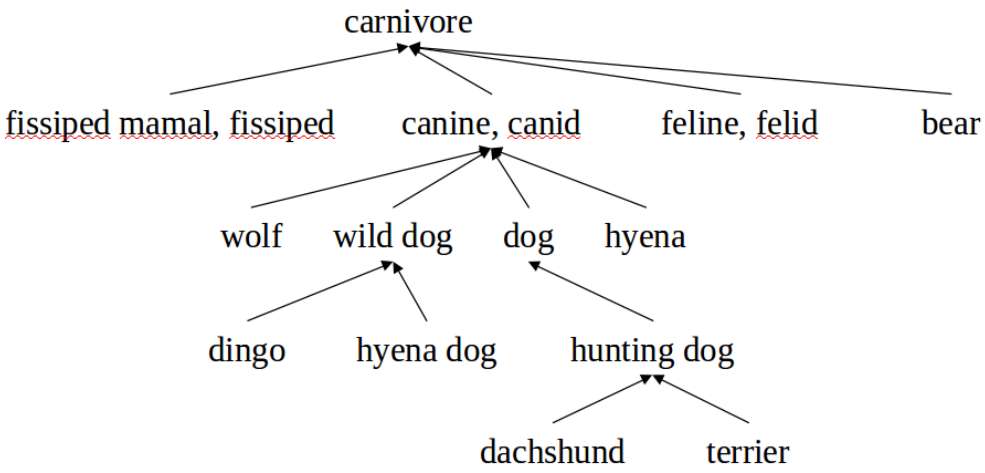
Given the selectional preference "DRINK BEVERAGE" : coffee#1

Knowledge-based WSD

- Semantic Similarity
- Words in a discourse must be related in meaning, for the discourse to be coherent (Haliday and Hassan, 1976)
- Use this property for WSD - Identify related meanings for words that share a common context

Knowledge-based WSD

- Semantic Similarity in a Local Context
- Similarity determined between pairs of concepts, or between a word and its surrounding context
- Relies on similarity metrics on semantic networks (Rada et al. 1989)



Knowledge-based WSD

- Semantic Similarity Metrics
- Input: two concepts (same part of speech)
- Output: similarity measure (Leacock and Chodorow 1998)

$$Similarity(C_1, C_2) = -\log\left(\frac{Path(C_1, C_2)}{2D}\right)$$, D is the taxonomy depth

- E.g. Similarity(wolf,dog) = 0.60 Similarity(wolf,bear) = 0.42

Knowledge-based WSD

- Semantic Similarity Metrics for WSD
- Disambiguate target words based on similarity with one word to the left and one word to the right (Patwardhan, Banerjee, Pedersen 2002)
- Example: disambiguate PLANT in "plant with flowers"
- PLANT:
  1. plant, works, industrial plant
  2. plant, flora, plant life
- Similarity (plant#1, flower) = 0.2
- Similarity (plant#2, flower) = 1.5

Knowledge-based WSD

- Heuristic-based Methods
- Example: “plant/flora” is used more often than “plant/factory”
- annotate any instance of PLANT as “plant/flora”
- Word meanings exhibit a Zipfian distribution

Knowledge-based WSD

- Most Frequent Sense
- Method 1: Find the most frequent sense in an annotated corpus
- Method 2: Find the most frequent sense using a method based on distributional similarity (McCarthy et al. 2004)

Knowledge-based WSD

- Most Frequent Sense
- Word senses
  pipe #1 = tobacco pipe
  pipe #2 = tube of metal or plastic
- Distributional similar words
  N = tube, cable, wire, tank, hole, cylinder, fitting, tap,...
- For each word in N, find similarity with pipe#i (using the sense that maximizes the similarity)
  pipe#1 - tube (#3) = 0.3 pipe#2 - tube (#1) = 0.6
- Compute score for each sense pipe#i
  score (pipe#1) = 0.25
  score (pipe#2) = 0.73
- Note: results depend on the corpus used to find distributionally similar words

Supervised Methods of Word Sense Disambiguation

- What is Supervised Learning?
- Collect a set of examples that illustrate the various possible classifications or outcomes of an event.
- Identify patterns in the examples associated with each particular class of the event.
- Generalize those patterns into rules.
- Apply the rules to classify a new event.

Supervised Methods of Word Sense Disambiguation

- Supervised WSD: Class of methods that induces a classifier from manually sense-tagged text using machine learning techniques.
- Resources:
  - Sense Tagged Text
  - Dictionary (implicit source of sense inventory)
  - Syntactic Analysis (POS tagger, Chunker, Parser, . . .)
- Scope
  - Typically one target word per context
  - Part of speech of target word resolved
- Reduces WSD to a classification problem where a target word is assigned the most appropriate sense from a given set of possibilities based on the context in which it occurs

Sense Tagged Text

- Sense Tagged Text

Bonnie and Clyde are two really famous criminals, I think they were **bank/1** robbers
My **bank/1** charges too much for an overdraft.
I went to the **bank/1** to deposit my check and get a new ATM card.
The University of Minnesota has an East and a West **Bank/2** campus right on the Mississippi River.
My grandfather planted his pole in the **bank/2** and got a great big catfish!
The **bank/2** is pretty muddy, I can't walk there.

Sense Tagged Text

- Two Bags of Words (Co-occurrences in the "window of context")

FINANCIAL_BANK_BAG:
a an and are ATM Bonnie card charges check Clyde criminals deposit famous for get I much My new overdraft really robbers the they think to too two went were
RIVER_BANK_BAG:
a an and big campus cant catfish East got grandfather great has his I in is Minnesota Mississippi muddy My of on planted pole pretty right River The the there University walk West

Simple Supervised Approach

- Given a sentence S containing "bank":

For each word W_i in S
If W_i is in FINANCIAL_BANK_BAG then
Sense_1 = Sense_1 + 1;
If W_i is in RIVER_BANK_BAG then
Sense_2 = Sense_2 + 1;

If Sense_1 > Sense_2 then print "Financial"
else if Sense_2 > Sense_1 then print "River"
else print "Can't Decide";

Supervised Methodology

- Create a sample of training data where a given target word is manually annotated with a sense from a predetermined set of possibilities.
  - One tagged word per instance/lexical sample disambiguation
- Select a set of features with which to represent context.
  - co-occurrences, collocations, POS tags, verb-obj relations, etc...
- Convert sense-tagged training instances to feature vectors.
- Apply a machine learning algorithm to induce a classifier.
- Convert a held out sample of test data into feature vectors.
  - "correct" sense tags are known but not used
- Apply classifier to test instances to assign a sense tag.

From Text to Feature Vectors

- My/pronoun grandfather/noun used/verb to/prep fish/verb along/adv the/det banks/SHORE of/prep the/det Mississippi/noun River/noun. (S1)
- The/det bank/FINANCE issued/verb a/det check/noun for/prep the/det amount/noun of/prep interest/noun. (S2)

	P-2	P-1	P+1	P+2	fish	check	river	interest	SENSE TAG
S1	adv	det	prep	det	Y	N	Y	N	SHORE
S2		det	verb	det	N	Y	N	Y	FINANCE

Supervised Learning Algorithms

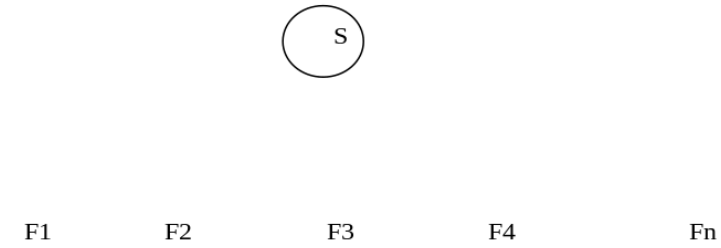
- Once data is converted to feature vector form, any supervised learning algorithm can be used. Many have been applied to WSD with good results:
  - Support Vector Machines
  - Nearest Neighbor Classifiers
  - Decision Trees
  - Decision Lists
  - Naive Bayesian Classifiers
  - Neural Networks
  - Linear Models

Naive Bayesian Classifier

- Naive Bayesian Classifier well known in Machine Learning community for good performance across a range of tasks (e.g., Domingos and Pazzani, 1997)
- Assumes conditional independence among features, given the sense of a word.
- The form of the model is assumed, but parameters are estimated from training instances
- When applied to WSD, features are often "a bag of words" that come from the training data
- Usually thousands of binary features that indicate if a word is present in the context of the target word (or not)

Naive Bayesian Model

- Bayesian Inference
- Given observed features, what is most likely sense?
- Estimate probability of observed features given sense
- Estimate unconditional probability of sense
- Unconditional probability of features is a normalizing term, doesn't affect sense classification



$$P(F1, F2, \ldots, Fn|S) = p(F1|S) * p(F2|S) * \ldots * p(Fn|S)$$

Naive Bayesian Classifier

- NB Classifier

$$sense = \underset{sense \in S}{\operatorname{argmax}}\, p(F1|S) * \ldots * p(Fn|S) * p(S)$$

  - Given 2,000 instances of "bank", 1,500 for bank/1 (financial sense) and 500 for bank/2 (river sense)
    - P(S=1) = 1,500/2000 = .75
    - P(S=2) = 500/2,000 = .25
  - Given "credit" occurs 200 times with bank/1 and 4 times with bank/2.
    - P(F1="credit") = 204/2000 = .102
    - P(F1="credit"|S=1) = 200/1,500 = .133
    - P(F1="credit"|S=2) = 4/500 = .008
  - Given a test instance that has one feature "credit"
    - P(S=1|F1="credit") = .133*.75/.102 = .978
    - P(S=2|F1="credit") = .008*.25/.102 = .020

Naive Bayesian Classifier

- Comparative Results
- (Leacock, et. al. 1993) compared Naïve Bayes with a Neural Network and a Context Vector approach when disambiguating six senses of line...
- (Mooney, 1996) compared Naïve Bayes with a Neural Network, Decision Tree/List Learners, Disjunctive and Conjunctive Normal Form learners, and a perceptron when disambiguating six senses of line...
- (Pedersen, 1998) compared Naïve Bayes with Decision Tree, Rule Based Learner, Probabilistic Model, etc. when disambiguating line and 12 other words...
- All found that Naïve Bayesian Classifier performed as well as any of the other methods!

Supervised WSD with Individual Classifiers

- Many supervised Machine Learning algorithms have been applied to Word Sense Disambiguation, most work reasonably well. (Witten and Frank, 2000) is a great intro. to supervised learning.
- Features tend to differentiate among methods more than the learning algorithms.
- Good sets of features tend to include:
- Co-occurrences or keywords (global)
- Collocations (local)
- Bigrams (local and global)
- Part of speech (local)
- Predicate-argument relations
- Verb-object, subject-verb,
- Heads of Noun and Verb Phrase

References

- Speech and Language Processing (3rd ed. draft) by D. Jurafsky & J. H. Martin (web.stanford.edu)