CS674 Natural Language Processing

- Last class
  - Metaphor, metonymy
  - Synonymy, hyponymy
  - Lexical semantic resources: WordNet
  - Word sense disambiguation: intro
- Today
  - Word sense disambiguation
    - Supervised learning
    - Issues for WSD evaluation

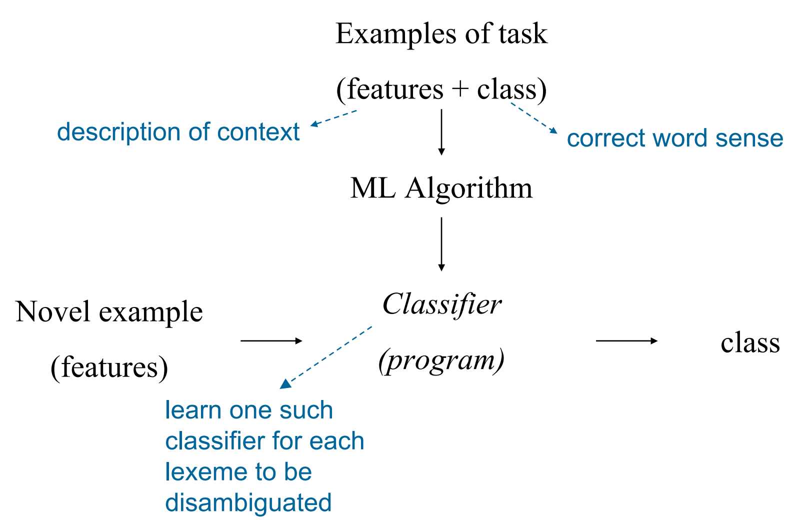
Word sense disambiguation

- Given a *fixed* set of senses is associated with a lexical item, determine which of them applies to a particular instance of the lexical item
- Two fundamental approaches
  - WSD occurs during semantic analysis as a side-effect of the elimination of ill-formed semantic representations
  - Stand-alone approach
    - WSD is performed independent of, and prior to, compositional semantic analysis
    - Makes minimal assumptions about what information will be available from other NLP processes
    - Applicable in large-scale practical applications

Machine learning approaches

- Machine learning methods
  - Supervised inductive learning
  - Bootstrapping
  - Unsupervised
- Emphasis is on acquiring the knowledge needed for the task from data, rather than from human analysts.

Inductive ML framework



Running example

*An electric guitar and **bass** player stand off to one side, not really part of the scene, just as a sort of nod to gringo expectations perhaps.*

1 Fish sense

2 Musical sense

3 …

Feature vector representation

- **target:** the word to be disambiguated
- **context** : portion of the surrounding text
  - Select a "window" size
  - Tagged with part-of-speech information
  - Stemming or morphological processing
  - Possibly some partial parsing
- Convert the context (and target) into a set of features
  - Attribute-value pairs
    - Numeric or nominal values

Collocational features

- Encode information about the lexical inhabitants of *specific* positions located to the left or right of the target word.
  - E.g. the word, its root form, its part-of-speech
  - *An electric <u>guitar and</u> **bass** <u>player stand</u> off to one side, not really part of the scene, just as a sort of nod to gringo expectations perhaps.*
  - [guitar, NN1, and, CJC, player, NN1, stand, VVB]

Co-occurrence features

- Encodes information about neighboring words, ignoring exact positions.
  - **Features**: the words themselves (or their roots)
  - **Values**: number of times the word occurs in a region surrounding the target word
  - Select a small number of frequently used content words for use as features
    - 12 most frequent content words from a collection of *bass* sentences drawn from the WSJ: *fishing, big, sound, player, fly, rod, pound, double, runs, playing, guitar, band*
    - Co-occurrence vector (window of size 10) for the previous example:
      [0,0,0,1,0,0,0,0,0,0,1,0]

Naïve Bayes classifiers for WSD

- Assumption: choosing the best sense for an input vector amounts to choosing the most probable sense for that vector

$$\hat{s} = \arg\max_{s \in S} P(s \mid V)$$

  – S denotes the set of senses
  – V is the context vector
- Apply Bayes rule:

$$\hat{s} = \arg\max_{s \in S} \frac{P(V \mid s) P(s)}{P(V)}$$

Naïve Bayes classifiers for WSD

- Estimate P(V|s):

$$P(V \mid s) \approx \prod_{j=1}^{\#\,feature-value\ pairs} P(v_j \mid s)$$

- P(*s*): proportion of each sense in the sense-tagged corpus

$$\hat{s} = \arg\max_{s \in S} P(s) \prod_{j=1}^{\#\,feature-value\ pairs} P(v_j \mid s)$$

- Mooney (1996) reports on *line* corpus that naïve-Bayes and an ANN worked best, achieving 73% correct.

WSD Evaluation

- Corpora:
  – *line* corpus
  – Yarowsky's 1995 corpus
    » 12 words (plant, space, bass, …)
    » ~4000 instances of each
  – Ng and Lee (1996)
    » 121 nouns, 70 verbs (most frequently occurring/ambiguous); WordNet senses
    » 192,800 occurrences
  – SEMCOR (Landes et al. 1998)
    » Portion of the Brown corpus tagged with WordNet senses
  – SENSEVAL (Kilgarriff and Rosenzweig, 2000)
    » Annual performance evaluation conference
    » Provides an evaluation framework (Kilgarriff and Palmer, 2000)
- Baseline: most frequent sense

WSD Evaluation

- Metrics
  – Precision
    » Nature of the senses used has a huge effect on the results
    » E.g. results using coarse distinctions cannot easily be compared to results based on finer-grained word senses
  – Partial credit
    » Worse to confuse musical sense of *bass* with a fish sense than with another musical sense
    » Exact-sense match → full credit
    » Select the correct broad sense → partial credit
    » Scheme depends on the organization of senses being used

Decision list classifiers

- Decision lists: equivalent to simple case statements.
  - Classifier consists of a sequence of tests to be applied to each input example/vector; returns a word sense.
- Continue only until the first applicable test.
- Default test returns the majority sense.

Decision list example

- Binary decision: fish *bass* vs. musical *bass*

Rule		Sense
fish within window	⇒	**bass**[1]
striped bass	⇒	**bass**[1]
guitar within window	⇒	**bass**[2]
bass player	⇒	**bass**[2]
piano within window	⇒	**bass**[2]
tenor within window	⇒	**bass**[2]
sea bass	⇒	**bass**[1]
play/V bass	⇒	**bass**[2]
river within window	⇒	**bass**[1]
violin within window	⇒	**bass**[2]
salmon within window	⇒	**bass**[1]
on bass	⇒	**bass**[2]
bass are	⇒	**bass**[1]

Learning decision lists

- Consists of *generating* and *ordering* individual tests based on the characteristics of the training data
- Generation: every feature-value pair constitutes a test
- Ordering: based on accuracy on the training set

$$abs\left(\log \frac{P(Sense_1 \mid f_i = v_j)}{P(Sense_2 \mid f_i = v_j)}\right)$$

- Associate the appropriate sense with each test