Today
- Lexical semantic resources: WordNet
- Word sense disambiguation
  » Dictionary-based approaches
  » Supervised machine learning methods
  » Issues for WSD evaluation

**Word sense disambiguation**

- Given a *fixed* set of senses 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

**Dictionary-based approaches**

- Rely on machine readable dictionaries
- Initial implementation of this kind of approach is due to Michael Lesk (1986)
  - Given a word \( W \) to be disambiguated in context \( C \)
    » Retrieve all of the sense definitions, \( S \), for \( W \) from the MRD
    » Compare each \( s \) in \( S \) to the dictionary definitions \( D \) of all the remaining words \( c \) in the context \( C \)
    » Select the sense \( s \) with the most overlap with \( D \) (the definitions of the context words \( C \))

**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

Examples of task (features + class)

description of context

ML Algorithm

Novel example (features)

Classifier (program)

learn one such classifier for each lexeme to be disambiguated

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, boolean, categorical, ...

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. ...

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 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.

<table>
<thead>
<tr>
<th>pre2-word</th>
<th>pre2-pos</th>
<th>pre1-word</th>
<th>pre1-pos</th>
<th>fol1-word</th>
<th>fol1-pos</th>
<th>fol2-word</th>
<th>fol2-pos</th>
</tr>
</thead>
<tbody>
<tr>
<td>guitar</td>
<td>NN1</td>
<td>and</td>
<td>CJC</td>
<td>player</td>
<td>NN1</td>
<td>stand</td>
<td>VVB</td>
</tr>
</tbody>
</table>
Co-occurrence features

- Encodes information about neighboring words, ignoring exact positions.
  - 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)
- Attributes: the words themselves (or their roots)
- Values: number of times the word occurs in a region surrounding the target word

\[
\begin{array}{cccccccccc}
\text{fishing?} & \text{big?} & \text{sound?} & \text{player?} & \text{fly?} & \text{rod?} & \text{pound?} & \text{double?} & \ldots & \text{guitar?} & \text{band?} \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0
\end{array}
\]

Inductive ML framework

Examples of task
(features + class)

description of context

ML Algorithm

Novel example
(features) ———
Classifier (program) ———
correct word sense

class

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

<table>
<thead>
<tr>
<th>Rule</th>
<th>Sense</th>
</tr>
</thead>
<tbody>
<tr>
<td>fish within window</td>
<td>bass¹</td>
</tr>
<tr>
<td>striped bass</td>
<td>bass¹</td>
</tr>
<tr>
<td>guitar within window</td>
<td>bass²</td>
</tr>
<tr>
<td>bass player</td>
<td>bass²</td>
</tr>
<tr>
<td>piano within window</td>
<td>bass²</td>
</tr>
<tr>
<td>tenor within window</td>
<td>bass²</td>
</tr>
<tr>
<td>tenor bass</td>
<td>bass¹</td>
</tr>
<tr>
<td>play'n bass</td>
<td>bass²</td>
</tr>
<tr>
<td>river within window</td>
<td>bass¹</td>
</tr>
<tr>
<td>viola within window</td>
<td>bass¹</td>
</tr>
<tr>
<td>trombone within window</td>
<td>bass²</td>
</tr>
<tr>
<td>on bass</td>
<td>bass¹</td>
</tr>
<tr>
<td>bass are</td>
<td>bass¹</td>
</tr>
</tbody>
</table>
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
  \[
  \text{abs} \left( \log \frac{P(Sense_1 \mid f_i = v_j)}{P(Sense_2 \mid f_i = v_j)} \right)
  \]
- Associate the appropriate sense with each test

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

CS474 Natural Language Processing

- **Before…**
  - Lexical semantic resources: WordNet
  - Word sense disambiguation
    - Dictionary-based approaches
- **Today**
  - Word sense disambiguation
    - Supervised machine learning methods
    - Weakly supervised (bootstrapping) methods
    - SENSEVAL
    - Unsupervised methods
Weakly supervised approaches

- **Problem:** Supervised methods require a large sense-tagged training set.
- **Bootstrapping approaches:** Rely on a small number of labeled *seed* instances.

### Generating initial seeds

- **Hand label a small set of examples**
  - Reasonable certainty that the seeds will be correct
  - Can choose prototypical examples
  - Reasonably easy to do
- **One sense per collocation** constraint (Yarowsky 1995)
  - Search for sentences containing words or phrases that are strongly associated with the target senses
    - Select *fish* as a reliable indicator of *bass_1*
    - Select *play* as a reliable indicator of *bass_2*
  - Or derive the collocations automatically from machine readable dictionary entries
  - Or select seeds automatically using collocational statistics (see Ch 6 of J&M)

### One sense per collocation

- **Yarowsky’s bootstrapping approach**
  - Relies on a **one sense per discourse** constraint: The sense of a target word is highly consistent within any given document
  - Evaluation on ~37,000 examples

<table>
<thead>
<tr>
<th>Word</th>
<th>Senses</th>
<th>Accuracy</th>
<th>Applicability</th>
</tr>
</thead>
<tbody>
<tr>
<td>plant</td>
<td>living/factory</td>
<td>99.8%</td>
<td>72.8%</td>
</tr>
<tr>
<td>tank</td>
<td>vehicle/container</td>
<td>99.6%</td>
<td>50.5%</td>
</tr>
<tr>
<td>poach</td>
<td>steal/boil</td>
<td>100.0%</td>
<td>44.4%</td>
</tr>
<tr>
<td>palm</td>
<td>tree/hand</td>
<td>99.8%</td>
<td>36.5%</td>
</tr>
<tr>
<td>axes</td>
<td>grid/tools</td>
<td>100.0%</td>
<td>36.5%</td>
</tr>
<tr>
<td>sake</td>
<td>ice/cool</td>
<td>100.0%</td>
<td>33.7%</td>
</tr>
<tr>
<td>bass</td>
<td>fish/music</td>
<td>100.0%</td>
<td>58.8%</td>
</tr>
<tr>
<td>space</td>
<td>volume/outer</td>
<td>99.2%</td>
<td>67.7%</td>
</tr>
<tr>
<td>motion</td>
<td>legal/physical</td>
<td>99.9%</td>
<td>49.8%</td>
</tr>
<tr>
<td>crane</td>
<td>bird/machine</td>
<td>100.0%</td>
<td>49.1%</td>
</tr>
<tr>
<td><strong>Average</strong></td>
<td></td>
<td>99.8%</td>
<td>50.1%</td>
</tr>
</tbody>
</table>
Yarowsky’s bootstrapping approach

To learn disambiguation rules for a polysemous word:

1. [Find all instances of the word in the training corpus and save the contexts around each instance.]

2. [For each word sense, identify a small set of training examples representative of that sense. Now we have a few labeled examples for each sense.]

3. Build a classifier (e.g. decision list) by training a supervised learning algorithm with the labeled examples.

4. Apply the classifier to all the unlabeled examples. Find instances that are classified with probability > a threshold and add them to the set of labeled examples.

5. Optional: Use the one-sense-per-discourse constraint to augment the new examples.

6. Go to Step 3. Repeat until the unlabelled data is stable.

SENSEVAL-2  2001

- Three tasks
  - Lexical sample
  - All-words
  - Translation

- 12 languages

- Lexicon
  - SENSEVAL-1: from HECTOR corpus
  - SENSEVAL-2: from WordNet 1.7

- 93 systems from 34 teams

CS474 Natural Language Processing

- Last class
  - Lexical semantic resources: WordNet
  - Word sense disambiguation
    » Dictionary-based approaches
    » Supervised machine learning methods

- Today
  - Word sense disambiguation
    » Supervised machine learning methods (finish)
    » Weakly supervised (bootstrapping) methods
      » SENSEVAL
    » Unsupervised methods

Lexical sample task

- Select a sample of words from the lexicon
- Systems must then tag instances of the sample words in short extracts of text

SENSEVAL-1: 35 words

- 700001 John Dos Passos wrote a poem that talked of ‘the <tag>bitter</tag> beat look, the scorn on the lip.”
- 700002 The beans almost double in size during roasting. Black beans are over roasted and will have a <tag>bitter</tag> flavour and insufficiently roasted beans are pale and give a colourless, tasteless drink.
## Lexical sample task: SENSEVAL-1

<table>
<thead>
<tr>
<th>Nouns</th>
<th>Verbs</th>
<th>Adjectives</th>
<th>Indeterminates</th>
</tr>
</thead>
<tbody>
<tr>
<td>-n</td>
<td>N</td>
<td>-v</td>
<td>N</td>
</tr>
<tr>
<td>accident</td>
<td>267</td>
<td>amaze</td>
<td>70</td>
</tr>
<tr>
<td>behaviour</td>
<td>279</td>
<td>bet</td>
<td>177</td>
</tr>
<tr>
<td>bet</td>
<td>274</td>
<td>bother</td>
<td>209</td>
</tr>
<tr>
<td>disability</td>
<td>160</td>
<td>bury</td>
<td>201</td>
</tr>
<tr>
<td>excess</td>
<td>186</td>
<td>calculate</td>
<td>217</td>
</tr>
<tr>
<td>float</td>
<td>75</td>
<td>consume</td>
<td>186</td>
</tr>
<tr>
<td>giant</td>
<td>118</td>
<td>derive</td>
<td>216</td>
</tr>
</tbody>
</table>
| ... | ...   | ... | ... | ... | ...
| TOTAL | 2756 | TOTAL | 2501 | TOTAL | 1406 | TOTAL | 1785 |

## All-words task

- Systems must tag almost all of the content words in a sample of running text
  - sense-tag all predicates, nouns that are heads of noun-phrase arguments to those predicates, and adjectives modifying those nouns
  - ~5,000 running words of text
  - ~2,000 sense-tagged words

## Translation task

- SENSEVAL-2 task
- Only for Japanese
- word sense is defined according to translation distinction
  - if the head word is translated differently in the given expressional context, then it is treated as constituting a different sense
- word sense disambiguation involves selecting the appropriate English word/phrase/sentence equivalent for a Japanese word

## SENSEVAL-2 results

<table>
<thead>
<tr>
<th>Language</th>
<th>Task</th>
<th>No. of submissions</th>
<th>No. of teams</th>
<th>IAA</th>
<th>Baseline</th>
<th>Best system</th>
</tr>
</thead>
<tbody>
<tr>
<td>Czech</td>
<td>AW</td>
<td>1</td>
<td>1</td>
<td>-</td>
<td>-</td>
<td>94</td>
</tr>
<tr>
<td>Basque</td>
<td>LS</td>
<td>3</td>
<td>2</td>
<td>.75</td>
<td>.65</td>
<td>76</td>
</tr>
<tr>
<td>Estonian</td>
<td>AW</td>
<td>2</td>
<td>2</td>
<td>.72</td>
<td>.85</td>
<td>67</td>
</tr>
<tr>
<td>Italian</td>
<td>LS</td>
<td>2</td>
<td>2</td>
<td>-</td>
<td>-</td>
<td>39</td>
</tr>
<tr>
<td>Korean</td>
<td>LS</td>
<td>2</td>
<td>2</td>
<td>-</td>
<td>.71</td>
<td>74</td>
</tr>
<tr>
<td>Spanish</td>
<td>LS</td>
<td>12</td>
<td>5</td>
<td>.64</td>
<td>.48</td>
<td>65</td>
</tr>
<tr>
<td>Swedish</td>
<td>LS</td>
<td>8</td>
<td>5</td>
<td>.95</td>
<td>-</td>
<td>70</td>
</tr>
<tr>
<td>Japanese</td>
<td>LS</td>
<td>7</td>
<td>3</td>
<td>.86</td>
<td>.72</td>
<td>78</td>
</tr>
<tr>
<td>Japanese</td>
<td>TL</td>
<td>9</td>
<td>8</td>
<td>.81</td>
<td>.37</td>
<td>79</td>
</tr>
<tr>
<td>English</td>
<td>AW</td>
<td>21</td>
<td>12</td>
<td>.75</td>
<td>.57</td>
<td>69</td>
</tr>
<tr>
<td>English</td>
<td>LS</td>
<td>26</td>
<td>15</td>
<td>.86</td>
<td>.51/16</td>
<td>64/40</td>
</tr>
</tbody>
</table>
SENSEVAL-2 de-briefing

- Where next?
  - Supervised ML approaches worked best
    » Looking at the role of feature selection algorithms
  - Need a well-motivated sense inventory
    » Inter-annotator agreement went down when moving to WordNet senses
  - Need to tie WSD to real applications
    » The translation task was a good initial attempt

SENSEVAL-3 2004

- 14 core WSD tasks including
  - All words (Eng, Italian): 5000 word sample
  - Lexical sample (7 languages)
- Tasks for identifying semantic roles, for multilingual annotations, logical form, subcategorization frame acquisition

English lexical sample task

- Data collected from the Web from Web users
- Guarantee at least two word senses per word
- 60 ambiguous nouns, adjectives, and verbs
- test data
  - ½ created by lexicographers
  - ½ from the web-based corpus
- Senses from WordNet 1.7.1 and Wordsmyth (verbs)
- Sense maps provided for fine-to-coarse sense mapping
- Filter out multi-word expressions from data sets

<table>
<thead>
<tr>
<th>Class</th>
<th>Nr of words</th>
<th>Avg senses (fine)</th>
<th>Avg senses (coarse)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nouns</td>
<td>20</td>
<td>5.8</td>
<td>4.35</td>
</tr>
<tr>
<td>Verbs</td>
<td>32</td>
<td>6.31</td>
<td>4.59</td>
</tr>
<tr>
<td>Adjectives</td>
<td>5</td>
<td>10.2</td>
<td>9.8</td>
</tr>
<tr>
<td>Total</td>
<td>57</td>
<td>6.47</td>
<td>4.96</td>
</tr>
</tbody>
</table>

Table 1: Summary of the sense inventory
Results

- 27 teams, 47 systems
- Most frequent sense baseline
  - 55.2% (fine-grained)
  - 64.5% (coarse)
- Most systems significantly above baseline
  - Including some unsupervised systems
- Best system
  - 72.9% (fine-grained)
  - 79.3% (coarse)

SENSEVAL-3 lexical sample results

<table>
<thead>
<tr>
<th>System/Team</th>
<th>Description</th>
<th>P</th>
<th>R</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>iTBuner (Thamburilliam et al.)</td>
<td>A supervised system using a Levenshtein similarity between context of ambiguous words and dictionary definitions. Experiments are performed for various window sizes, various similarity measures</td>
<td>66.1</td>
<td>65.7</td>
<td>74.9</td>
</tr>
<tr>
<td>ITT Kernel</td>
<td>Random methods for sense prediction, pattern mining and syntactic analysis, and unsupervised term induction in an SVM classifier.</td>
<td>73.6</td>
<td>72.6</td>
<td>79.3</td>
</tr>
<tr>
<td>mostl</td>
<td>A combination of knowledge sources (post-of-words of neighboring words, words in context, local collocations, syntactic relations), in an SVM classifier.</td>
<td>71.4</td>
<td>72.4</td>
<td>78.8</td>
</tr>
<tr>
<td>TAU</td>
<td>Similar to iTBuner, with different correction function of a post-of-words</td>
<td>72.3</td>
<td>72.3</td>
<td>78.8</td>
</tr>
<tr>
<td>BUC</td>
<td>A combination of knowledge sources (post-of-words of neighboring words, words in context, local collocations, syntactic relations), in a SVM classifier.</td>
<td>71.4</td>
<td>72.4</td>
<td>78.8</td>
</tr>
<tr>
<td>BUC-Exp</td>
<td>Similar to BUC, but with a neural classifier.</td>
<td>71.4</td>
<td>72.4</td>
<td>78.8</td>
</tr>
</tbody>
</table>

SENSEVAL-3 results (unsupervised)

<table>
<thead>
<tr>
<th>System/Team</th>
<th>Description</th>
<th>P</th>
<th>R</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cameron (Un)</td>
<td>A maximum entropy model for unsupervised Shifting, using neighboring words and syntactic structures as features. A few annotated instances are used to train context classes to WordNet/WordNet sense</td>
<td>56.3</td>
<td>56.3</td>
<td>64.4</td>
</tr>
<tr>
<td>PML</td>
<td>An unsupervised system, using local part-of-speech and frequency information</td>
<td>54.7</td>
<td>54.7</td>
<td>63.6</td>
</tr>
<tr>
<td>CL-Research (Likhosher)</td>
<td>An unsupervised system solving on definition properties, syntactic, semantic, subcategorization patterns, other lexical information, as given in a dictionary.</td>
<td>45.0</td>
<td>45.0</td>
<td>55.5</td>
</tr>
<tr>
<td>CASSENKO</td>
<td>A maximum entropy model for unsupervised Shifting, using neighboring words and syntactic structures as features. A few annotated instances are used to train context classes to WordNet/WordNet sense</td>
<td>50.1</td>
<td>41.7</td>
<td>49.3</td>
</tr>
</tbody>
</table>

CS474 Natural Language Processing

- Last class
  - Lexical semantic resources: WordNet
  - Word sense disambiguation
    » Dictionary-based approaches
    » Supervised machine learning methods

- Today
  - Word sense disambiguation
    » Supervised machine learning methods (finish)
    » Issues for WSD evaluation
    » SENSEVAL
    » Weakly supervised (bootstrapping) methods
    Unsupervised methods
Unsupervised WSD

- Rely on **agglomerative clustering** to cluster feature-vector representations (without class/word-sense labels) according to a similarity metric.
- Represent each cluster as the average of its constituent feature-vectors.
- Label the cluster by hand with known word senses.
- Unseen feature-encoded instances are classified by assigning the word sense of the most similar cluster.
- Schuetze (1992, 1998) uses a (complex) clustering method for WSD.
  - For coarse binary decisions, unsupervised techniques can achieve results approaching those of supervised and bootstrapping methods.
  - In most cases approaching the 90% range.
  - Tested on a small sample of words.

Issues for evaluating clustering

- The correct senses of the instances used in the training data may not be known.
- The clusters are almost certainly heterogeneous w.r.t. the sense of the training instances contained within them.
- The number of clusters is almost always different from the number of senses of the target word being disambiguated.