Idea: Lexical Affinity Models

- Words select other words on syntactic grounds

\[
\text{congress narrowly passed the amended bill}
\]

- Idea: Link up pairs with high mutual information
  - [Yuret, 1998]: Greedy linkage
  - [Paskin, 2001]: Iterative re-estimation with EM

- Evaluation: compare linked pairs to a gold standard

Problem: Non-Syntactic Affinity

- Mutual information between words does not necessarily indicate syntactic selection.

\[
\text{congress narrowly passed the amended bill}
\]
\[
\text{expect brushbacks but no beanballs}
\]
\[
\text{a new year begins in new york}
\]

A Word-Class Model

\[
P(\text{words, classes, graph}) = P(\text{length}) P(\text{graph})
\]

Idea: Word Classes

- Individual words like congress are enwined with semantic facts about the world.
- Syntactic classes, like NOUN and ADVERB are bleached of word-specific semantics.
- Automatic word classes more likely to look like DAYS-OF-WEEK or PERSON-NAME.
- We could build dependency models over word classes.
  - [cf. Carroll and Charniak, 1992]
Problems: Word Class Models

- Issues:
  - Too simple a model – doesn’t work much better supervised
  - No representation of valence (number of arguments)

<table>
<thead>
<tr>
<th>Model</th>
<th>Random</th>
<th>Carroll and Charniak, 92</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>41.7</td>
<td>44.7</td>
</tr>
</tbody>
</table>

NOUN NOUN VERB
stock prices fell

NOUN NOUN VERB
stock prices fell

A Head-Outward Model (DMV)

- Supervised statistical parsers benefit from modeling tree distributions implicitly. [e.g., Collins, 99]
- A head-outward model with word classes and valence/adjacency:

$$P(t_i) = \prod_{d \in \{r, l\}} \quad \text{arg}$$

**Results: Dependencies**

- Model is re-estimated with EM
- Cubic dynamic program run over each sentence
- Expected counts of each modeled configuration are aggregated
- Initialization:
- Initial parameters from simple heuristics:

$$P(a \mid h, \text{dir}) \propto \sum_{\text{dist}} \frac{1}{\text{dist}} \text{count}(h, a, \text{dist}, \text{dir})$$

Common Errors: Dependency

<table>
<thead>
<tr>
<th>Overproposed Dependencies</th>
<th>Underproposed Dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>DT → NN</td>
<td>DET → N</td>
</tr>
<tr>
<td>NNP → NNP</td>
<td>NNP → NNP</td>
</tr>
<tr>
<td>CD → CD</td>
<td>NN → NNS</td>
</tr>
<tr>
<td>IN ← DT</td>
<td>NN → VBP</td>
</tr>
<tr>
<td>DT ← NNS</td>
<td>DT ← NNS</td>
</tr>
<tr>
<td>DT → IN</td>
<td>NN ← IN</td>
</tr>
<tr>
<td>DT → VBD</td>
<td>DT → VBP</td>
</tr>
<tr>
<td>DT → VBP</td>
<td>NN → VBD</td>
</tr>
</tbody>
</table>

Early Approaches: Structure Search

- Incremental grammar learning, chunking [Wolff B8, Langley B2, many others]
- Can recover synthetic grammars
- An (extremely good) result of incremental structure search:

- Looks good, … but can’t parse in the wild.
Issues with Chunk/Merge Systems

- Hard to recover from initial choices (c.f. EM, where the issue is initial state)
- Hard to make local decisions which will interact well with each other (e.g. group verb-preposition and preposition-determiner, both wrong, and not consistent)
- Good local heuristics often don’t have a well-formed global objective that can be evaluated for the target grammar.

Idea: Learn PCFGs with EM

- Classic experiments on learning PCFGs with Expectation-Maximization [Lari and Young, 1990]

\[
\{ X_1, X_2, \ldots, X_k \} \quad \begin{array}{c}
X_i \\
\ldots \\
X_k
\end{array}
\]

- Full binary grammar over \( n \) symbols
- Parse uniformly/randomly at first
- Re-estimate rule expectations off of parses
- Repeat
- Their conclusion: it doesn’t really work.

Re-estimation of PCFGs

- Basic quantity needed for re-estimation with EM:

\[
P(X_{ij} | i, j, S) = \frac{\sum_{T \models S} P(T)}{\sum_{T \models S} P(T)}
\]

- Can calculate in cubic time with the Inside-Outside algorithm.
- Consider an initial grammar where all productions have equal weight:

\[
P(X_{ij} | X_j, X_i) = \frac{1}{n^2}
\]

- Then all trees have equal probability initially.
- Therefore, after one round of EM, the posterior over trees will (in the absence of random perturbation) be approximately uniform over all trees, and symmetric over symbols.

Problem: “Uniform” Posteriors

- Tree Uniform
- Split Uniform

Problem: Model Symmetries

- Symmetries
- How does this relate to trees?

Other Approaches

- Evaluation: fraction of nodes in gold trees correctly posited in proposed trees (unlabeled recall)
- Some recent work in learning constituency:
  - [Adrians, 99] Language grammars aren’t general PCFGs
  - [Clark, 01] Mutual-information filters detect constituents, then an MDL-guided search assembles them
  - [van Zaanen, 00] Finds low edit-distance sentence pairs and extracts their differences
Right-Branching Baseline

- English trees tend to be right-branching, not balanced
- A simple (English-specific) baseline is to choose the right chain structure for each sentence

Desiderata: Practical Learnability

- To be practically learnable, models should:
  - Be as simple as possible
  - Make symmetries self-breaking whenever possible
  - Avoid hidden structures which are not directly coupled to surface phenomena

Inspiration: Distributional Clustering

- the president said that the downturn was over

Distributional Models

\[ P(S, C) = \prod_i P(c_i)P(w_i | c_i)P(w_{i+1} | w_i, c_i) \]

Problem: Identifying Constituents

Distributional classes are easy to find...

... but figuring out which are constituents is hard.
**A Nested Distributional Model**

- We’d like a model that:
  - Ties spans to linear contexts (like distributional clustering)
  - Considers only proper tree structures (like a PCFG model)
  - Has no symmetries to break (like a dependency model)

**Constituent-Context Model (CCM)**

\[
P(S|T) = \prod_{(i,j) \in T} P(\text{factory payrolls fell in september})
\]

**Results: Constituency**

- Right-Branch: 70.0

**Spectrum of Systematic Errors**

- CCM analysis better
- Treebank analysis better

**Results: Combined Models**

- Supervised (but unannotated) PCFG constituency recall is at 92.8
- Qualitative improvements
  - Subject-verb groups gone, modifier placement improved

**How General is This?**

<table>
<thead>
<tr>
<th>Language</th>
<th>Constituency Evaluation</th>
<th>Dependency Evaluation</th>
</tr>
</thead>
<tbody>
<tr>
<td>English (7422 sentences)</td>
<td>CCM+DMV: 89.7</td>
<td>CCM: 69.7</td>
</tr>
<tr>
<td>Random Baseline</td>
<td>39.4</td>
<td>45.6</td>
</tr>
<tr>
<td>CCM+DMV</td>
<td>89.7</td>
<td>54.2</td>
</tr>
<tr>
<td>German (2175 sentences)</td>
<td>CCM+DMV: 89.7</td>
<td>CCM: 69.7</td>
</tr>
<tr>
<td>Random Baseline</td>
<td>39.4</td>
<td>45.6</td>
</tr>
<tr>
<td>CCM+DMV</td>
<td>89.7</td>
<td>54.2</td>
</tr>
<tr>
<td>Chinese (2473 sentences)</td>
<td>CCM+DMV: 89.7</td>
<td>CCM: 69.7</td>
</tr>
<tr>
<td>Random Baseline</td>
<td>39.4</td>
<td>45.6</td>
</tr>
<tr>
<td>CCM+DMV</td>
<td>89.7</td>
<td>54.2</td>
</tr>
</tbody>
</table>
### Most Common Errors: English

**Overproposed Constituents**

<table>
<thead>
<tr>
<th>JJ NN</th>
<th>1022</th>
<th>the [general partner]</th>
</tr>
</thead>
<tbody>
<tr>
<td>NNP NNP</td>
<td>447</td>
<td>the [Big Board]</td>
</tr>
<tr>
<td>DT NN</td>
<td>398</td>
<td>[an import] order</td>
</tr>
<tr>
<td>JJ NNS</td>
<td>294</td>
<td>six million [common shares]</td>
</tr>
<tr>
<td>NNS RB</td>
<td>164</td>
<td>seats currently</td>
</tr>
</tbody>
</table>

**Crossing Constituents**

| CD CD IN CD CD | 154 | rose to [billion from billion] |
| NNS RB         | 133 | petroleum [prices also] surged |
| NNP NNP NNP    | 67  | to [Hong Kong China] is      |
| JJ NN          | 66  | especially [strong growth]   |

### Most Common Errors: German

**Overproposed Constituents**

| JJ NN     | 461  | der [Dalberger Hof] |
| DT NN     | 430  | [die Moderatoren] der Zukunft |
| DT JJ NN  | 94   | Aus [der erhofften Meisterschaft] |
| CC NN     | 71   | Sim [und Roma] |

**Crossing Constituents**

| JJ NN     | 30   | New [Yorker Aktienbourse] |
| CD CD     | 18   | 300 [000 Mark] |
| NNP NNP   | 17   | Frankfurt A. [M. FR] |
| IN NN     | 15   | [zwischen Schwips und Kater] |

### Recap

- **Unsupervised learning:**
  - Constituency structure
  - Dependency structure

- **Constituency recall:**

<table>
<thead>
<tr>
<th>Model</th>
<th>Recall</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random Baseline</td>
<td>39.4</td>
</tr>
<tr>
<td>CCM + DMV</td>
<td>88.0</td>
</tr>
<tr>
<td>Supervised PCFG</td>
<td>92.8</td>
</tr>
</tbody>
</table>

- **Why it works:**
  - Combination of simple models
  - Representations designed for unsupervised learning