Intrinsic and Extrinsic Evaluations

[Galliers and Sparck Jones (1993)]

- **Intrinsic Evaluation Criteria:**
  - Relating to a system’s objective

- **Extrinsic Evaluation Criteria:**
  - Relating to the system’s function i.e. to its role in relation to its setup’s purpose

---

Intrinsic versus Extrinsic Evaluations of Parsing Systems

Diego Mollá-Allod
Ben Hutchinson
14 April 2003

- Link Grammar and Conexor FDG
- Intrinsic Evaluation
- Extrinsic Evaluation: Answer Extraction
- Discussion

---

Link Grammar and Conexor FDG

- Two examples of state-of-the-art parsing systems
- Robust treatment of “difficult” or “ungrammatical” sentences
- Dependency-based

- **Link Grammar:**
  - Publicly available
  - Developed by Carnegie Mellon University
  - [http://www.link.cs.cmu.edu/link/](http://www.link.cs.cmu.edu/link/)

- **Conexor Functional Dependency Grammar (Conexor FDG):**
  - Proprietary
  - Initially developed by the University of Helsinki
  - [http://www.conexor.fi/](http://www.conexor.fi/)

---

Link Grammar and Conexor FDG

- **Link Grammar**

  \[
  \begin{align*}
  &\text{the man} < \text{subj} \\
  &\text{that} < \text{det} \\
  &\text{came} < \text{ins} \\
  &\text{ate} < \text{pcomp} < \text{det} \\
  &\text{v} < \text{subj} \\
  &\text{bananas} < \text{obj} < \text{cc} \\
  &\text{apples} < \text{obj} < \text{cc} \\
  &\text{with} < \text{pcomp} < \text{det} \\
  &\text{a} < \text{det} \\
  &\text{fork} < \text{pcomp} < \text{det} \\
  \end{align*}
  \]

- **Conexor FDG**

  \[
  \begin{align*}
  &\text{main} < \text{subj} \\
  &\text{sub} < \text{det} \\
  &\text{mod} < \text{det} \\
  &\text{ins} < \text{ins} \\
  &\text{cc} < \text{cc} \\
  &\text{pcomp} < \text{pcomp} < \text{det} \\
  \end{align*}
  \]
Intrinsic Evaluation — Grammatical Relations

Grammatical Relations

- The man that came ate bananas and apples with a fork.
  (detmod _ man the) (cmod that man come) (nsubj come man _) (nsubj eat man _) (obj eat banana _) (obj eat apple _) (conj and banana apple) (ncmod fork eat with) (detmod _ fork a)

- Same example with the selected grmrels
  (mod that man come) (subj come man _) (subj eat man _) (obj eat banana _) (obj eat apple _) (mod fork eat with)

Intrinsic Evaluation

- Failure to do this will continue to place a disproportionate burden on Fulton taxpayers.
  (xcomp to failure do) (obj do this _) (nsubj continue failure _) (xcomp to continue place) (nsubj place failure _) (obj place burden _) (nmod _ burden disproportionate) (iobj on place tax-payer) (nmod _ tax-payer Fulton) (detmod _ burden a) (aux _ continue will)

- Same example with the selected grmrels
  (xcomp to failure do) (obj do this _) (subj continue failure _) (xcomp to continue place) (subj place failure _) (obj place burden _) (mod _ burden disproportionate) (iobj on place tax-payer) (mod _ tax-payer Fulton)
Results of Intrinsic Evaluation

<table>
<thead>
<tr>
<th></th>
<th>With Link Grammar</th>
<th>With Conexor FDG</th>
</tr>
</thead>
<tbody>
<tr>
<td>Precision</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SUBJ</td>
<td>50.3%</td>
<td>73.6%</td>
</tr>
<tr>
<td>OBJ</td>
<td>48.5%</td>
<td>84.8%</td>
</tr>
<tr>
<td>XCOMP</td>
<td>62.2%</td>
<td>76.2%</td>
</tr>
<tr>
<td>MOD</td>
<td>57.2%</td>
<td>63.7%</td>
</tr>
<tr>
<td>Average</td>
<td>54.6%</td>
<td>74.6%</td>
</tr>
<tr>
<td>Recall</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SUBJ</td>
<td>39.1%</td>
<td>64.5%</td>
</tr>
<tr>
<td>OBJ</td>
<td>50%</td>
<td>53.4%</td>
</tr>
<tr>
<td>XCOMP</td>
<td>32.1%</td>
<td>64.7%</td>
</tr>
<tr>
<td>MOD</td>
<td>53.7%</td>
<td>56.2%</td>
</tr>
<tr>
<td>Average</td>
<td>43.7%</td>
<td>59.7%</td>
</tr>
</tbody>
</table>

Extrinsic Evaluation

- Embedding setup: Answer Extraction
  - Locate those exact phrases of unedited text documents that answer a query worded in natural language
- ExTraNS
  - An answer extraction system
  - Uses logical forms to determine the answer of a question
  - The version for the present evaluation uses:
    - a full parser (Link Grammar or Conexor FDG);
    - a semantic interpreter;
    - a simple thesaurus based on WordNet;
    - an answer extraction module that operates on logical forms.

The Logical Forms

- Called Minimal Logical forms because they encode the minimum information required for AE
- Flat expressions that use reification

Example: *cp will quickly copy files*

```
holds(e4, object(cp, o1, [x1])), object(s_command, o2, [x1]),
evt(s_copy, e4, [x1, x6]), object(s_file, o3, [x6]), prop(quickly, p3, [e4]).
```

Example: *the man that came ate bananas and apples with a fork*

```
holds(e1, object(s_man, o2, [x2])), evt(s_come, e4, [x2]),
evt(s_eat, e5, [x7]), e6@<e7, e8@<e7, evt(s_eat, e5_1, [x6]),
evt(s_eat, e5_2, [x8]), object(s_banana, o6, [x6]), object(s_apple, o8, [x8]), prop(with, p9, [e6]),
object(s_fork, o11, [x11]).
```

Scoring the Overlap of Logical Forms

- **Synonym mode:**
  - Find the synonym representatives
  - Use Prolog resolution
  - Only finds exact matches
- **Approximate mode:**
  - Find the synonym representatives
  - Compute the highest overlap possible with variable unification
  - Return the sentence(s) with highest overlap
  - If there are exact matches, Synonym mode and Approximate mode return the same answers
Extrinsic Evaluation

- Corpus:
  - 500 Unix manual pages
  - 26 sample questions with the answers found in the corpus
- Nature of the questions:
  - There is at least one answer in the corpus
  - The question asks how to perform a particular action, or how a particular command works
  - The question is simple
- Precision and recall as in standard Information Retrieval
- F-score = 2 (|Returned and relevant|) / (|Returned| + |Relevant|)

Results of Extrinsic Evaluation

<table>
<thead>
<tr>
<th>Synonym mode</th>
<th>Precision</th>
<th>Recall</th>
<th>F-score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parser</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Conexor FDG</td>
<td>55.8%</td>
<td>0.9%</td>
<td>0.074</td>
</tr>
<tr>
<td>LG-best</td>
<td>49.7%</td>
<td>11.4%</td>
<td>0.099</td>
</tr>
<tr>
<td>LG-all</td>
<td>50.9%</td>
<td>13.1%</td>
<td>0.120</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Approximate mode</th>
<th>Precision</th>
<th>Recall</th>
<th>F-score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parser</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Conexor FDG</td>
<td>28.3%</td>
<td>21.9%</td>
<td>0.177</td>
</tr>
<tr>
<td>LG-best</td>
<td>31.8%</td>
<td>15.8%</td>
<td>0.150</td>
</tr>
<tr>
<td>LG-all</td>
<td>40.5%</td>
<td>20.5%</td>
<td>0.183</td>
</tr>
</tbody>
</table>

Final Discussion: Intrinsic or Extrinsic Evaluations?

- A "good" parser is not necessarily best for an application?
  - The conversion to grammatical relations may throw away important information
  - Consistent errors/idioms in the parser output can be corrected in subsequent processing stages
  - John wanted.v Mary to.o come.v
  - Variables introduced in the evaluation may affect the results...

To Do

- Extrinsic evaluation where AE is based on the overlap of grammatical relations
  - To remove variables in the experiments
- Use same corpus for both intrinsic and extrinsic evaluations
  - Any suggestions?
- Intrinsic evaluation of parser+semantic interpreter
- Use other intrinsic evaluations of parsers (e.g. constituency-based)
- Use other embedding setups for extrinsic evaluations
  - To test if similar results occur