Are UD Treebanks Getting More Consistent?  
A Report Card for English UD  
AMIR ZELDES & NATHAN SCHNEIDER  
{amir.zeldes,nathan.schneider}@georgetown.edu

Overview
- Universal Dependencies (UD) provides 200+ treebanks in 138 languages with a unified scheme (de Marneffe et al. 2021)
- 40/138 languages have multiple treebanks, allowing joint models
- English default for popular tools uses EWT+GUM (Stanza, Qi et al. 2020)
- But treebanks are not necessarily consistent and constantly changing

How consistent are English EWT and GUM? Where do they differ?
- Is consistency improving across UD versions? (focus on v2.6-2.12)
- Is joint training for English a good idea? If so, since when?

Where do GUM and EWT trees still differ?
- Proper name internal structure (incl. conversion errors)
- Some compounds
- Number modifiers
- List markers (LS)
- Dates
- Deprel list (almost unused in GUM)

Parsing experiments
- Is cross-corpus parsing getting better?
  - Methodology: fix GUM train to 2.6 documents (GUM has grown since)
  - Use Diaparser (Attardi et al. 2021) + Electra (Clark et al. 2020)

What do parsers get wrong? (more in paper!)
- Cross-corpus results are getting better but are worse than within-corpus
- GUM is harder (-1.4 LAS at best); 12 genres incl. spoken, less data

How has the data changed?
- Methodology: treat each successive version as gold and the previous as pred
- Use official CoNLL scorer to obtain delta to next version to quantify change

Table 1: Cross-corpus parsing scores (three run averages with standard deviations)

<table>
<thead>
<tr>
<th>Version</th>
<th>LAS</th>
<th>Lemmas</th>
<th>Tokens</th>
<th>UPOS</th>
<th>XPOS</th>
</tr>
</thead>
<tbody>
<tr>
<td>EWT</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>v2.6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>v2.7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>v2.8</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>v2.9</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>v2.10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GUM</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>v2.6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>v2.7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>v2.8</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>v2.9</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>v2.10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Bottom line: in realistic usage on new data use joint models!!

Scores for joint model have gotten steadily better
- Still can’t beat train/test on single same corpus!
- But macro-average on both corpora is much better
- And the gap is now very small even within-corpus (best joint model less than 0.5 LAS away from best within-corpus model for both corpora)

Bottom line: in realistic usage on new data use joint models!!

Where to find this work:
- Sixth Workshop on Universal Dependencies (UDW2023)
- Sixth Workshop on Universal Dependencies (UDW2023)