Neural Constituency Parsing

Berkeley NLP

Dan Klein
CS 288
She enjoys playing tennis.
She enjoys playing tennis.
Historical Trends

[Slide from Slav Petrov]
She enjoys playing tennis.
Grammars

\[ S \rightarrow \text{NP} \ \text{VP} \]

\[ \text{NP}^S \rightarrow she \]

\[ \text{VP[enjoys]} : \text{S[playing]} \]

\[
\begin{array}{c}
\text{S} \\
\text{NP} \\
\text{She} \\
\text{VP} \\
\text{enjoys} \\
\text{S} \\
\text{VP} \\
\text{playing} \\
\text{NP} \\
\text{tennis}
\end{array}
\]
She enjoys playing tennis.
She enjoys playing tennis.
She enjoys playing tennis.
Routing with LSTMs

She enjoys playing tennis.
Routing with LSTMs

She enjoys playing tennis.
Routing with LSTMs

Verb at the start

She enjoys playing tennis.
Routing with LSTMs

She enjoys playing tennis.
Routing with LSTMs

She enjoys playing tennis.
Span Classification

She enjoys playing tennis.
She enjoys playing tennis.
She enjoys playing tennis.
She enjoys playing tennis.
She enjoys playing tennis.
Non-Constituents

Ø

She enjoys playing tennis.

VP

She enjoys playing tennis.
She enjoys playing tennis.
She enjoys playing tennis.
Does It Work?

<table>
<thead>
<tr>
<th>Method</th>
<th>F1 Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Grammar-Based [Carreras et al, 08]</td>
<td>91.0</td>
</tr>
<tr>
<td>LSTM-Based [Stern et al, 17]</td>
<td>92.6</td>
</tr>
</tbody>
</table>

F1 (English, dev)
Neural parsers no longer have much of the model structure provided to classical parsers. How do they perform so well without it?
What’s Going on in There?

Why don’t we need a grammar?

Adjacent tree labels are redundant with LSTM features

If we can predict surrounding tree labels from our LSTM representation of the input, then this information doesn’t need to be provided explicitly by grammar production rules

We find that for 92.3% of spans, the label of the span’s parent can predicted from the neural representation of the span

```
<START> She played soccer in the park <STOP>
```
What’s Going on in There?

Do we need tree constraints?

Not for F1

Many neural parsers no longer model output correlations with grammar rules, but still use output correlations from tree constraints.

Predicting span brackets independently gives nearly identical performance on PTB development set F1 and produces valid trees for 94.5% of sentences.
What’s Going on in There?

Is distant context important?

Almost a full point of F1 is lost by truncating context 5 words away from span endpoints and half a point with 10 words.
**What’s Going on in There?**

<table>
<thead>
<tr>
<th>Do LSTMs introduce useful inductive bias compared to feedforward networks?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yes!</td>
</tr>
<tr>
<td>We compare a truncated LSTM with feedforward architectures that are given the same inputs</td>
</tr>
<tr>
<td>The LSTM outperformed the best feedforward by <strong>6.5 F1</strong></td>
</tr>
</tbody>
</table>
Routing with Transformers

Query: verb

She enjoys playing tennis.
<table>
<thead>
<tr>
<th>She</th>
<th>enjoys</th>
<th>playing</th>
<th>tennis</th>
<th>.</th>
</tr>
</thead>
</table>

**Query:** verb

- verb [VBZ]
- verb [VBG]
- noun
- punctuation
She enjoys playing tennis.
She enjoys playing tennis.
She enjoys playing tennis.
She enjoys playing tennis.
What Helps?

LSTM
Self-Attentive
+Factored

F1 (English, dev)
Results: Multilingual

- Arabic
  - Björkelund et al. (2014): 81.3
  - Coavoux and Crabbé (2017): 82.9
  - Cross and Huang (2016): 85.6
  - Ours: 87.5

- German
  - Björkelund et al. (2014): 81.7
  - Coavoux and Crabbé (2017): 85.3
  - Cross and Huang (2016): 86.0
  - Ours: 87.7

- Hebrew
  - Björkelund et al. (2014): 89.8
  - Coavoux and Crabbé (2017): 89.9
  - Cross and Huang (2016): 90.4
  - Ours: 90.4

- Korean
  - Björkelund et al. (2014): 81.7
  - Coavoux and Crabbé (2017): 85.3
  - Cross and Huang (2016): 87.7
  - Ours: 89.7

- Polish
  - Björkelund et al. (2014): 90.5
  - Coavoux and Crabbé (2017): 91.2
  - Cross and Huang (2016): 93.6
  - Ours: 93.7

- Swedish
  - Björkelund et al. (2014): 84.0
  - Coavoux and Crabbé (2017): 84.5
  - Cross and Huang (2016): 85.5
  - Ours: 85.5
Pre-Training

Problem: Input has more variation than output

Need to handle:
• Rare words not seen during training
• Word forms in morphologically rich languages
• Contextual paraphrase / lexical variation
Historical Trends

[Slide from Slav Petrov]
Knowledge Modularity

- Knowledge modularity: Learn domain-general knowledge from one data source and use it solve specific problems elsewhere
Key Idea: Embed contexts, not words. Use these embeddings for other tasks.

Example: BERT (Devlin et al., 2019) -- bidirectional Transformer trained on masked language modeling and next-sentence prediction
Explosion of Pretraining Work

GLUE Baseline (ICLR 2019)

- ALBERT (Ensemble)
- ALICE v2 large ensemble (Alibaba DAMO NLP)
- FreeLB-RoBERTa (ensemble)
- RoBERTa
- XLNet-Large (ensemble)
- MT-DNN-ensemble
- GLUE Human Baselines
- Snorkel MeTaL
- XLM (English only)
- SemBERT
- SpanBERT (single-task training)
- BERT + BAM
- Span-Extractive BERT on STILTs
- BERT on STILTs
- RGLM-Base (Huawei Noah's Ark Lab)
- BERT: 24-layers, 16-heads, 1024-hidden
- BERT + Single-task Adapters
- Macaron Net-base
- SesameBERT-Base
- MobileBERT
- StackingBERT-Base
- TinyBERT
- BILSTM+ELMo+Attn

GLUE SoTA (ICLR 2020)

- ELMo
- ULMFit
- BERT
- RoBERTa
- MT-DNN
- MASS
- UniLM
- XLM
- SpanBERT
- Cross-lingual
- Multi-task
- +Generation
- Knowledge distillation
- Span-prediction
- Remove NSP
- Longer time
- Remove NSP
- More data
- Permutation LM
- Transformer-XL
- More data
- +Knowledge Graph
- Cross-modal
- Insertion-based Generation

KERMIT

GPT

GPT-2

VideoBERT
- CBT
- ViLBERT
- VisualBERT
- B2T2
- Unicoder-VL
- LXMBERT
- VL-BERT
- UNITER

KERNIT

BIDIRECTIONAL LM

Whole-Word Masking

Human

Defense

ERNE (Baidu)

BERT-wwm

By Xiaochi Wang & Zhengyuan Zhang @THUNLP
She enjoys playing tennis.
She enjoys playing tennis.
She enjoys playing tennis.
Encoder Architectures

<table>
<thead>
<tr>
<th></th>
<th>No pre-training</th>
<th>Pre-training (with ELMo)</th>
<th>Pre-training (with BERT)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>LSTM</strong></td>
<td>92.08 F1</td>
<td>95.13 F1</td>
<td>95.60 F1</td>
</tr>
<tr>
<td></td>
<td>[Gaddy+ 2018]</td>
<td>[Kitaev &amp; Klein 2018]</td>
<td>[Kitaev et al. 2019]</td>
</tr>
<tr>
<td><strong>Self-Attention</strong></td>
<td>93.55 F1</td>
<td>95.60 F1</td>
<td></td>
</tr>
</tbody>
</table>
Encoder Architectures

F1 Score (English)

No pre-training: 93.6
ELMo: 95.2
BERT-base: 95.3
BERT-large: 95.6
XLNet-large: 96.0

Number of Parameters

No pre-training: 26M
ELMo: 107M
BERT-base: 117M
BERT-large: 343M
XLNet-large: 361M
Results: Multilingual

- Arabic:
  - Björkelund et al. (2014): 81.3, 82.9, 85.6, 87
  - Kitaev and Klein (2018): 88.0
  - Coavoux and Crabbé (2017): 88.2, 88.8, 89.7, 91
  - This work (one model per language): 89.0
  - This work (joint multilingual model):

- Basque:
  - Björkelund et al. (2014): 88.2, 88.8, 89.7, 92
  - Coavoux and Crabbé (2017): 88.2, 88.8, 89.7, 92
  - This work (one model per language): 89.0
  - This work (joint multilingual model):

- German:
  - Björkelund et al. (2014): 81.7, 85.3, 87.7, 90
  - Kitaev and Klein (2018): 85.3
  - Coavoux and Crabbé (2017): 89.8, 90.4, 93
  - This work (one model per language):

- Hebrew:
  - Björkelund et al. (2014): 89.8, 90.4, 93
  - Coavoux and Crabbé (2017): 89.8, 90.4, 93
  - This work (one model per language):

- Polish:
  - Björkelund et al. (2014): 90.5, 93.6, 96
  - Kitaev and Klein (2018): 90.5, 93.6, 96
  - Coavoux and Crabbé (2017): 90.5, 93.6, 96
  - This work (one model per language):

- Hungarian:
  - Björkelund et al. (2014): 88.2, 88.8, 89.7, 92
  - Coavoux and Crabbé (2017): 89.7, 92.7, 95
  - This work (one model per language):

- Korean:
  - Björkelund et al. (2014): 81.7, 86.6, 89
  - Kitaev and Klein (2018): 89
  - This work (one model per language):

- Swedish:
  - Björkelund et al. (2014): 80.5, 82.5, 84, 85.5
  - Kitaev and Klein (2018): 82.5
  - Coavoux and Crabbé (2017): 84.0, 84.5
  - This work (one model per language):
Figure 1: Labelled bracketing F1 versus minimum span length for the English corpora. F1 scores for the In-Order parser with BERT (orange) and the Chart parser with BERT (cyan) start to diverge for longer spans.
Out of Domain Parsing

<table>
<thead>
<tr>
<th></th>
<th>Berkeley F1</th>
<th>Δ Err.</th>
<th>BLLIP F1</th>
<th>Δ Err.</th>
<th>In-Order F1</th>
<th>Δ Err.</th>
<th>Chart F1</th>
<th>Δ Err.</th>
</tr>
</thead>
<tbody>
<tr>
<td>WSJ Test</td>
<td>90.06</td>
<td>+0.0%</td>
<td>91.48</td>
<td>+0.0%</td>
<td>91.47</td>
<td>+0.0%</td>
<td>93.27</td>
<td>+0.0%</td>
</tr>
<tr>
<td>Brown All</td>
<td>84.64</td>
<td>+54.5%</td>
<td>85.89</td>
<td>+65.6%</td>
<td>85.60</td>
<td>+68.9%</td>
<td>88.04</td>
<td>+77.7%</td>
</tr>
<tr>
<td>Genia All</td>
<td>79.11</td>
<td>+110.2%</td>
<td>79.63</td>
<td>+139.1%</td>
<td>80.31</td>
<td>+130.9%</td>
<td>82.68</td>
<td>+157.4%</td>
</tr>
<tr>
<td>EWT All</td>
<td>77.38</td>
<td>+127.6%</td>
<td>79.91</td>
<td>+135.8%</td>
<td>79.07</td>
<td>+145.4%</td>
<td>82.22</td>
<td>+164.2%</td>
</tr>
</tbody>
</table>

Neural parsers improve out-of-domain numbers, but not more than in-domain numbers
Other Neural Constituency Parsers

- Back to at least Henderson 1998!
- Recent directions:
  - Shift-Reduce, eg Cross and Huang 2016
  - SR/Generative, eg Dyer et al 2016 (RNNG)
  - In-Order Generative, eg Liu and Zhang 2017
Open Source Release

Code and models are publicly available at: github.com/nikitakit/self-attentive-parser

Sample Usage (with spaCy integration)

```python
>>> import spacy
>>> from benepar.spacy_plugin import BeneparComponent
>>> nlp = spacy.load('en')
>>> nlp.add_pipe(BeneparComponent("benepar_en"))
>>> doc = nlp(u"Short cuts make long delays.")
>>> sent = list(doc.sents)[0]
>>> print(sent._.parse_string)
(S (NP (JJ Short) (NNS cuts)) (VP (VBP make) (NP (JJ long) (NNS delays))) (.) .)
>>> sent._.labels
('S',)
>>> list(sent._.children)[0]
Short cuts
```

Sample Usage (with NLTK integration)

```python
>>> import benepar
>>> parser = benepar.Parser("benepar_en")
>>> tree = parser.parse("Short cuts make long delays.")
>>> print(tree)
(S (NP (JJ Short) (NNS cuts)) (VP (VBP make) (NP (JJ long) (NNS delays))) (.) .))
```