Top-down Tree Long Short-Term Memory Networks

**Xingxing Zhang**, Liang Lu, Mirella Lapata

School of Informatics, University of Edinburgh

12th June, 2016
Sequential Language Models

\[ P(S = w_1, w_2, \ldots, w_n) = \prod_{i=1}^{n} P(w_i|w_1:i-1) \]  

- State of the Art
  - based on Long Short Term Memory Network Language Model (Hochreiter and Schmidhuber, 1997; Sundermeyer et al., 2012)
  - **Billion word benchmark results** reported in Jozefowicz et al., (2016)

<table>
<thead>
<tr>
<th>Models</th>
<th>PPL</th>
</tr>
</thead>
<tbody>
<tr>
<td>KN5</td>
<td>67.6</td>
</tr>
<tr>
<td>LSTM</td>
<td>30.6</td>
</tr>
<tr>
<td>LSTM+CNN INPUTS</td>
<td>30.0</td>
</tr>
</tbody>
</table>
Will tree structures help LMs?

Zhang et al., 2016
Will tree structures help LMs?

- Probably yes
  - LMs based on Constituency Parsing (Chelba and Jelinek, 2000; Roark, 2001; Charniak, 2001)
  - LMs based on Dependency Parsing (Shen et al., 2008; Zhang, 2009; Sennrich, 2015)
LSTMs + Dependency Trees = TreeLSTMs

Why?
- Sentence Length $N$ v.s. Tree Height $\log(N)$
LSTMs + Dependency Trees = TreeLSTMs

- Why?
  - Sentence Length $N$ v.s. Tree Height $\log(N)$

- How?
  - Top-down Generation
  - Breadth-first search
  - reminiscent of Eisner (1996)
The luxury auto manufacturer last year sold 1,214 cars in the U.S.

ROOT
The luxury auto manufacturer last year sold 1,214 cars in the U.S.
The luxury auto manufacturer last year sold 1,214 cars in the U.S.

```
ROOT
  ↓
sold
  ↓
year
```
The luxury auto manufacturer last year sold 1,214 cars in the U.S.
The luxury auto manufacturer last year sold 1,214 cars in the U.S.
The luxury auto manufacturer last year sold 1,214 cars in the U.S.
The luxury auto manufacturer last year sold 1,214 cars in the U.S.
Tree LSTM

\[ P(S) = \prod_{i=1}^{n} P(w_i|w_{1:i-1}) \quad (2) \]

\[ P(S|T) = \prod_{w \in \text{BFS}(T) \setminus \text{ROOT}} P(w|D(w)) \quad (3) \]

- \( D(w) \) is the *Dependency Path* of \( w \).
- \( D(w) \) is a generated sub-tree.
- Works on *projective* and *unlabeled* dependency trees.
Tree LSTM

Zhang et al., 2016
Tree LSTM

Zhang et al., 2016
Tree LSTM
Tree LSTM
Tree LSTM

Zhang et al., 2016
Tree LSTM

Generated by four LSTMs with tied $W_e$ and tied $W_{ho}$.

- GEN-Nx-L
- GEN-Nx-R

Zhang et al., 2016
One Limitation of Tree LSTM

Zhang et al., 2016

The luxury auto last 1,214 in the U.S.}

The manufacturer year cars in U.S. sold
Left Dependent Tree LSTM

Zhang et al., 2016
Left Dependent Tree LSTM
Left Dependent Tree LSTM
Left Dependent Tree LSTM
Experiments
1) I have seen it on him, and could _____ to it.
   a) write  b) migrate  c) climb  d) swear  e) contribute

- Training set: 49 million words (around 2 million sentences)
- development set: 4000 sentences
- test set: 1040 completion questions.
<table>
<thead>
<tr>
<th>Method</th>
<th>Reference</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>ivLBL</td>
<td>Mnih and Kavukcuoglu, 2013</td>
<td>55.5</td>
</tr>
<tr>
<td>LBL</td>
<td>Mnih and Teh, 2012</td>
<td>54.7</td>
</tr>
<tr>
<td>LSTM</td>
<td>This work</td>
<td>57.02</td>
</tr>
<tr>
<td>BiLSTM</td>
<td>This work</td>
<td>49.90</td>
</tr>
<tr>
<td>Skip-gram + RNNMEs</td>
<td>Mikolov et al., 2013</td>
<td>58.9</td>
</tr>
<tr>
<td>Model</td>
<td>Method</td>
<td>Score</td>
</tr>
<tr>
<td>---------------------</td>
<td>-------------------------------</td>
<td>-------</td>
</tr>
<tr>
<td>IVLBL</td>
<td>Mnih and Kavukcuoglu, 2013</td>
<td>55.5</td>
</tr>
<tr>
<td>LBL</td>
<td>Mnih and Teh, 2012</td>
<td>54.7</td>
</tr>
<tr>
<td>LSTM</td>
<td>This work</td>
<td>57.02</td>
</tr>
<tr>
<td>BiLSTM</td>
<td>This work</td>
<td>49.90</td>
</tr>
<tr>
<td>Skip-gram + RNNMEs</td>
<td>Mikolov et al., 2013</td>
<td>58.9</td>
</tr>
<tr>
<td>TreeLSTM</td>
<td>Our model</td>
<td>56.73</td>
</tr>
<tr>
<td>LdTreeLSTM</td>
<td>Our model</td>
<td>60.67</td>
</tr>
</tbody>
</table>
Dependency Parsing Reranking

- Rerank 2nd Order MSTParser (McDonald and Pereira, 2006)
- We train TreeLSTM and LdTreeLSTM as language models.
- We only use words as input features; POS tags, dependency labels or composition features are not used.
Dependency Parsing Reranking

NN: Chen & Manning, 2014; S-LSTM: Dyer et al., 2015
Dependency Parsing Reranking

NN: Chen & Manning, 2014; S-LSTM: Dyer et al., 2015

Zhang et al., 2016
Dependency Parsing Reranking

NN: Chen & Manning, 2014; S-LSTM: Dyer et al., 2015

Zhang et al., 2016

Tree LSTM

12th June, 2016 14 / 18
Dependency Parsing Reranking

NN: Chen & Manning, 2014; S-LSTM: Dyer et al., 2015
Tree Generation

Four binary classifiers:
- Add Left? No!

Features: hidden states and word embeddings

<table>
<thead>
<tr>
<th>Classifiers</th>
<th>Accuracies</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add-Left</td>
<td>94.3</td>
</tr>
<tr>
<td>Add-Right</td>
<td>92.6</td>
</tr>
<tr>
<td>Add-Nx-Left</td>
<td>93.4</td>
</tr>
<tr>
<td>Add-Nx-Right</td>
<td>96.0</td>
</tr>
</tbody>
</table>
Tree Generation

Four binary classifiers:
- Add Right? Yes!

Features: hidden states and word embeddings

<table>
<thead>
<tr>
<th>Classifiers</th>
<th>Accuracies</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add-Left</td>
<td>94.3</td>
</tr>
<tr>
<td>Add-Right</td>
<td>92.6</td>
</tr>
<tr>
<td>Add-Nx-Left</td>
<td>93.4</td>
</tr>
<tr>
<td>Add-Nx-Right</td>
<td>96.0</td>
</tr>
</tbody>
</table>
Tree Generation

Four binary classifiers:
- Add Right? Yes!

Features: hidden states and word embeddings

<table>
<thead>
<tr>
<th>Classifiers</th>
<th>Accuracies</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add-Left</td>
<td>94.3</td>
</tr>
<tr>
<td>Add-Right</td>
<td>92.6</td>
</tr>
<tr>
<td>Add-Nx-Left</td>
<td>93.4</td>
</tr>
<tr>
<td>Add-Nx-Right</td>
<td>96.0</td>
</tr>
</tbody>
</table>
Tree Generation

Four binary classifiers:
- Add Next Right? No!
- Add Next Left?
- Add Right?
- Add Next Right?

Features: hidden states and word embeddings

<table>
<thead>
<tr>
<th>Classifiers</th>
<th>Accuracies</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add-Left</td>
<td>94.3</td>
</tr>
<tr>
<td>Add-Right</td>
<td>92.6</td>
</tr>
<tr>
<td>Add-Nx-Left</td>
<td>93.4</td>
</tr>
<tr>
<td>Add-Nx-Right</td>
<td>96.0</td>
</tr>
</tbody>
</table>

Zhang et al., 2016

Tree LSTM
Tree Generation

Four binary classifiers:
- Add Left? Yes!

Features: hidden states and word embeddings

<table>
<thead>
<tr>
<th>Classifiers</th>
<th>Accuracies</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add-Left</td>
<td>94.3</td>
</tr>
<tr>
<td>Add-Right</td>
<td>92.6</td>
</tr>
<tr>
<td>Add-Nx-Left</td>
<td>93.4</td>
</tr>
<tr>
<td>Add-Nx-Right</td>
<td>96.0</td>
</tr>
</tbody>
</table>

Zhang et al., 2016
Tree Generation

Four binary classifiers:
- Add Left? Yes!

Features: hidden states and word embeddings

<table>
<thead>
<tr>
<th>Classifiers</th>
<th>Accuracies</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add-Left</td>
<td>94.3</td>
</tr>
<tr>
<td>Add-Right</td>
<td>92.6</td>
</tr>
<tr>
<td>Add-Nx-Left</td>
<td>93.4</td>
</tr>
<tr>
<td>Add-Nx-Right</td>
<td>96.0</td>
</tr>
</tbody>
</table>
Tree Generation

Four binary classifiers:
- Add Next Left? No!
- Add Left?
- Add Right?
- Add Next Left?
- Add Next Right?

Features: hidden states and word embeddings

<table>
<thead>
<tr>
<th>Classifiers</th>
<th>Accuracies</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add-Left</td>
<td>94.3</td>
</tr>
<tr>
<td>Add-Right</td>
<td>92.6</td>
</tr>
<tr>
<td>Add-Nx-Left</td>
<td>93.4</td>
</tr>
<tr>
<td>Add-Nx-Right</td>
<td>96.0</td>
</tr>
</tbody>
</table>
Tree Generation

Four binary classifiers:
- Add Left?
- Add Right?
- Add Next Left?
- Add Next Right?

Features: hidden states and word embeddings

<table>
<thead>
<tr>
<th>Classifiers</th>
<th>Accuracies</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add-Left</td>
<td>94.3</td>
</tr>
<tr>
<td>Add-Right</td>
<td>92.6</td>
</tr>
<tr>
<td>Add-Nx-Left</td>
<td>93.4</td>
</tr>
<tr>
<td>Add-Nx-Right</td>
<td>96.0</td>
</tr>
</tbody>
</table>
Profit widened to $\text{UNK}$ million, from $1.37$ billion a year earlier.

But Mr. O’Kicki said all industry executives certainly do n’t have to focus now.

That would postpone a stock activity in the forefront of the monetary policy.
Conclusions

- Syntax can help language modeling.
- Predicting tree structures with Neural Networks is possible.
- Next Steps:
  - Sequence to Tree Models
  - Tree to Tree Models
- code available:
  https://github.com/XingxingZhang/td-treelstm

Thanks & Questions?