Prepositional Phrase Attachment over Word Embedding Products

*Pranava Madhyastha*\(^{(1)}\), Xavier Carreras\(^{(2)}\), Ariadna Quattoni\(^{(2)}\)

\(^{(1)}\) University of Sheffield

\(^{(2)}\) Naver Labs Europe
Prepositional Phrases

I went to the restaurant by bike v/s I went to the restaurant by the bridge
Prepositional Phrases

I went to the restaurant by bike

v/s

I went to the restaurant by the bridge
I went to the restaurant by bike

v/s

I went to the restaurant by the bridge
Prepositional Phrases

![Diagram of prepositional phrases]

- \( \text{I went to the restaurant by bike} \)
- \( \text{v/s I went to the restaurant by the bridge} \)

- \( s/\text{bridge}/\text{Arno}/ \Rightarrow ? \)
Local PP Attachment Decisions

- Ratnaparkhi; 1998 approached PP attachment as a local classification decision
- Lose potential advantage for inference over the whole structure
Ratnaparkhi; 1998 approached PP attachment as a local classification decision
Lose potential advantage for inference over the whole structure
Two settings:
- Binary attachment problem[Ratnaparkhi; 1998]
- Multi-class attachment problem[Belinkov+; 2015]
Word Embeddings

- Previous approaches: features in a binary decision setting
- Word Embeddings are rich source of information
- How far can we go with only word embeddings?
The problem reduces to finding compatibility between: \( \phi(\text{head}) \), \( \phi(\text{prep}) \) and \( \phi(\text{mod}) \).

One simple approach is to do:

\[
f(h, p, m) = \vec{w} \cdot [v_h \otimes v_p \otimes v_m]
\]

Exploit each dimension alone - with a simple trick: append 1.
Our Proposal: Unfolding the Tensors

This results in:

\[ f(h, p, m) = v_h^\top \ W [v_p \otimes v_m] \]

We can further explore compositionality as:

\[ f(h, p, m) = \alpha(h)^\top \ W \beta(p, m) \]
Training the Model

- We use logistic loss
- Std. conditional Max. Likelihood optimization

\[
\arg\min_{\mathbf{W}} \text{logistic}(\mathbf{Tset}, \mathbf{W}) + \lambda \|\mathbf{W}\|_*
\] (1)

- As a structural constraint, we use rank-constrained regularization
Empirical Overview

- Exploring Word Embeddings
- Exploring Compositionality
- Binary Attachment Datasets
- Multiple Attachment Datasets
- Semi-supervised multiple attachment experiments with PTB and WTB
Exploring Word Embeddings

- Word2Vec based skip-gram embeddings ([Mikolov+; 2013])
- Skip-dep ([Bansal+; 2014]): uses word2vec with dependency contexts
- Exploration over different dimensions and data.
## Exploring Word Embeddings

<table>
<thead>
<tr>
<th>Word Embedding</th>
<th>Source Data</th>
<th>Accuracy wrt. dimension ((n))</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>(n = 50)</td>
</tr>
<tr>
<td>Skip-gram</td>
<td>BLLIP</td>
<td>83.23</td>
</tr>
<tr>
<td>Skip-gram</td>
<td>Wikipedia</td>
<td>83.74</td>
</tr>
<tr>
<td>Skip-gram</td>
<td>NYT</td>
<td>84.76</td>
</tr>
<tr>
<td>Skip-dep</td>
<td>BLLIP</td>
<td>85.52</td>
</tr>
<tr>
<td>Skip-dep</td>
<td>Wikipedia</td>
<td>84.23</td>
</tr>
<tr>
<td>Skip-dep</td>
<td>NYT</td>
<td>85.27</td>
</tr>
<tr>
<td>Skip-gram &amp; Skip-dep</td>
<td>BLLIP</td>
<td></td>
</tr>
</tbody>
</table>

- Attachment accuracy over RRR devset
Exploring Compositionality

\[ f(h, p, m) = \alpha(h)^\top W \beta(p, m) \]

Exploring the possibilities with \( \beta(p, m) \):
Exploring Compositionality

\[ f(h, p, m) = \alpha(h)^T W \beta(p, m) \]

Exploring the possibilities with \( \beta(p, m) \):

\[ \Rightarrow \text{Sum : } f(h, p, m) = \alpha(h)^T W (w_p + w_m) \]
Exploring Compositionality

\[ f(h, p, m) = \alpha(h)^T W \beta(p, m) \]

Exploring the possibilities with \( \beta(p, m) \):

\[ \Rightarrow \text{Sum} : f(h, p, m) = \alpha(h)^T W (w_p + w_m) \]

\[ \Rightarrow \text{Concatenation} : f(h, p, m) = \alpha(h)^T \mathbf{W}(w_p; w_m) \]
Exploring Compositionality

\[ f(h, p, m) = \alpha(h)^T \mathbf{W} \beta(p, m) \]

Exploring the possibilities with \( \beta(p, m) \):

\[ \Rightarrow \text{Sum} : f(h, p, m) = \alpha(h)^T \mathbf{W}(w_p + w_m) \]
\[ \Rightarrow \text{Concatenation} : f(h, p, m) = \alpha(h)^T \mathbf{W}(w_p; w_m) \]
\[ \Rightarrow \text{Products} : f(h, p, m) = \alpha(h)^T \mathbf{W}(w_p \otimes w_m) \]
Exploring Compositionality

\[ f(h, p, m) = \alpha(h)^T \mathbf{W} \beta(p, m) \]

Exploring the possibilities with \( \beta(p, m) \) :

\[ \Rightarrow \text{Sum : } f(h, p, m) = \alpha(h)^T \mathbf{W}(w_p + w_m) \]

\[ \Rightarrow \text{Concatenation : } f(h, p, m) = \alpha(h)^T \mathbf{W}(w_p; w_m) \]

\[ \Rightarrow \text{Products : } f(h, p, m) = \alpha(h)^T \mathbf{W}(w_p \otimes w_m) \]

\[ \Rightarrow \text{Prep. Identities : } f(h, p, m) = \alpha(h)^T \mathbf{W} p w_m \]
Exploring Compositionality

<table>
<thead>
<tr>
<th>Composition of $p$ and $m$</th>
<th>Tensor Size</th>
<th>Acc.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sum</td>
<td>$[v_p + v_m]$</td>
<td>$n \times n$</td>
</tr>
<tr>
<td>Concatenation</td>
<td>$[v_p; v_m]$</td>
<td>$n \times 2n$</td>
</tr>
<tr>
<td>$p$ Identities</td>
<td>$[i_p \otimes v_m]$</td>
<td>$n \times</td>
</tr>
<tr>
<td>Product</td>
<td>$[v_p \otimes v_m]$</td>
<td>$n \times n \times n$</td>
</tr>
</tbody>
</table>
Empirical Results over Binary Attachment Datasets

<table>
<thead>
<tr>
<th>Method</th>
<th>Word Embedding</th>
<th>RRR</th>
<th>WIKI</th>
<th>NYT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tensor product</td>
<td>Skip-gram, Wikipedia, ( n = 100 )</td>
<td>84.96</td>
<td>83.48</td>
<td>82.13</td>
</tr>
<tr>
<td></td>
<td>Skip-gram, NYT, ( n = 100 )</td>
<td>85.11</td>
<td>83.52</td>
<td>82.65</td>
</tr>
<tr>
<td></td>
<td>Skip-dep, BLLIP, ( n = 100 )</td>
<td>86.13</td>
<td>83.60</td>
<td>82.30</td>
</tr>
<tr>
<td></td>
<td>Skip-dep, Wikipedia, ( n = 100 )</td>
<td>85.01</td>
<td>83.53</td>
<td>82.10</td>
</tr>
<tr>
<td></td>
<td>Skip-dep, NYT, ( n = 100 )</td>
<td>85.49</td>
<td>83.64</td>
<td>83.47</td>
</tr>
<tr>
<td>[Stetina+; 1997(*)]</td>
<td></td>
<td>88.1</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>[Collins+; 1999]</td>
<td></td>
<td>84.1</td>
<td>72.7</td>
<td>80.9</td>
</tr>
<tr>
<td>[Belinkov+; 2014(*)]</td>
<td></td>
<td>85.6</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>[Nakashole+; 2015(*)]</td>
<td></td>
<td>84.3</td>
<td>79.3</td>
<td>84.3</td>
</tr>
</tbody>
</table>

Scores refer to Accuracy measures
Multiple Heads and Positional Information

\[ f(h, p, m) = \alpha(h)^\top W \beta(p, m) \]

In this case, we can modify \( \alpha(h) \) to include the positional info:

- \( \alpha(h) = \delta_h \otimes v_h \), where \( \delta_h \in \mathbb{R}^{|H|} \) is the position vector

This can also be written as:

\[ f(h, p, m) = v_h^\top W \delta_h [v_p \otimes v_m] . \]
## Experiments with Multiple Attachment Setting

<table>
<thead>
<tr>
<th></th>
<th>Test Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Arabic</td>
</tr>
<tr>
<td>Tensor product ($n=50$, $\ell_2$)</td>
<td>-</td>
</tr>
<tr>
<td>Tensor product ($n=50$, $\ell_*$)</td>
<td>-</td>
</tr>
<tr>
<td>Tensor product ($n=100$, $\ell_*$)</td>
<td>81.1</td>
</tr>
<tr>
<td>Belinkov+; 2014 (basic)</td>
<td>77.1</td>
</tr>
<tr>
<td>Belinkov+; 2014 (syn)</td>
<td>79.1</td>
</tr>
<tr>
<td>Belinkov+; 2014 (feat)</td>
<td>80.4</td>
</tr>
<tr>
<td>Belinkov+; 2014 (full)</td>
<td>82.6</td>
</tr>
<tr>
<td>Yu+; 2016 (full)</td>
<td>-</td>
</tr>
</tbody>
</table>
Embeddings trained on *source* and *target raw-data*
Model trained on the source data
Compare against Stanford neural network based parser and Turbo parser
Semi-supervised Experiments with PTB vs WTB

<table>
<thead>
<tr>
<th></th>
<th>PTB Test (2523)</th>
<th>Web Treebank Test</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Tensor BLLIP</td>
<td><strong>89.0</strong></td>
<td>82.7</td>
<td>82.6</td>
</tr>
<tr>
<td>Tensor BLLIP+WTB</td>
<td>88.9</td>
<td><strong>83.3</strong></td>
<td><strong>85.2</strong></td>
</tr>
<tr>
<td>Chen+; 2014</td>
<td>87.3</td>
<td>79.3</td>
<td>79.7</td>
</tr>
<tr>
<td>Martins+; 2010 (2\textsuperscript{nd} order)</td>
<td>88.8</td>
<td>83.6</td>
<td>83.7</td>
</tr>
<tr>
<td>Martins+; 2010 (3\textsuperscript{rd} order)</td>
<td><strong>88.9</strong></td>
<td><strong>84.2</strong></td>
<td>84.5</td>
</tr>
</tbody>
</table>

Scores refer to Accuracy measures
What Seems to Work

Came the disintegration of the Beatles’ minds with LSD...
Where it has a Problem

... the return address for the letters to the Senators ...
Conclusions

- We described a PP attachment model using simple tensor products
- Product of embeddings seem to be better at capturing compositions
- In out-of-domain datasets, we see the models show a big promise
- Proposed models can serve as a building block to dependency parsing methods
Thank You
Learning with Structural Constraints

- Controlling the learned parameter matrix with regularization
Learning with Structural Constraints

- Controlling the learned parameter matrix with regularization
- We can obtain dense, dispersed parameter matrix with $\ell_2$
Learning with Structural Constraints

- Controlling the learned parameter matrix with regularization

\[
SVD(W) = U\Sigma V^\top
\]

\[
\begin{bmatrix}
  \mathbf{u}_{11} & \cdots & \mathbf{u}_{1k} \\
  \mathbf{u}_{21} & \cdots & \mathbf{w}_{2k} \\
  \vdots & \ddots & \vdots \\
  \mathbf{u}_{n1} & \cdots & \mathbf{u}_{nk}
\end{bmatrix}
\]

\[
\begin{bmatrix}
  \sigma_1 & \cdots & 0 \\
  \vdots & \ddots & \vdots \\
  0 & \cdots & \sigma_k
\end{bmatrix}
\]

\[
\begin{bmatrix}
  \mathbf{v}_{11} & \cdots & \cdots & \mathbf{v}_{1n} \\
  \vdots & \ddots & \ddots & \vdots \\
  \mathbf{v}_{k1} & \cdots & \cdots & \mathbf{v}_{kn}
\end{bmatrix}
\]
Learning with Structural Constraints

- Controlling the learned parameter matrix with regularization
- We can obtain dense, dispersed parameter matrix with $\ell_2$
- Obtain a low-rank matrix with $\ell_\star$ regularization
Learning with Structural Constraints

- Controlling the learned parameter matrix with regularization
- Bonus: Everything can be convex optimized with FOBOS!!!
A Quicklook at Optimization

Input: Gradient function $g$

1. $W_{t+1} = \arg\min_W ||W_{t+0.5} - W||^2_2 + \eta_t \lambda r(W)$

Output: $W_{t+1}$

2. $W_1 = 0$

3. while $t < T$ do

4. $\eta_t = \frac{c}{\sqrt{t}}$

5. $W_{t+0.5} = W_t - \eta_t g(W_t)$

6. if $\ell_2$ regularizer then

7. $W_{t+1} = \frac{1}{1 + \eta_t \lambda} W_{t+0.5}$

8. else if $\ell_1$ regularizer then

9. $U \Sigma V^\top = \text{SVD}(W_{t+0.5})$

10. $\tilde{\Sigma}_{i,i} = \max(\Sigma_{i,i} - \eta_t \lambda, 0)$

11. $W_{t+1} = U \tilde{\Sigma} V^\top$

end
BLLIP with \(\sim\)1.8 million sentences and \(\sim\)43 million tokens of Wall Street Journal text (and excludes PTB evaluation sets);

- English Wikipedia \(\sim\)13.1 million sentences and \(\sim\)129 million tokens;

- The New York Times portion of the GigaWord corpus, with \(\sim\)52 million sentences and \(\sim\)1,253 million tokens.