Discriminative Neural Sentence Modeling by Tree-Based Convolution

Lili Mou, Hao Peng, Ge Li, Yan Xu, Lu Zhang, Zhi Jin

Software Institute, Peking University, P. R. China

EMNLP, Lisbon, Portugal
September, 2015
Outline

1. Introduction & Related Work
2. Tree-Based Convolution
   - c-TBCNN
   - d-TBCNN
3. Experimental Results
   - Experiment I: Sentiment Analysis
   - Experiment II: Question Classification
   - Model Analysis
4. Conclusion
Outline

1. Introduction & Related Work

2. Tree-Based Convolution
   - c-TBCNN
   - d-TBCNN

3. Experimental Results
   - Experiment I: Sentiment Analysis
   - Experiment II: Question Classification
   - Model Analysis

4. Conclusion
Sentence modeling

- To capture the meaning of a sentence
- Related to various tasks in NLP [Kalchbrenner et al., 2014]
  - Sentiment analysis
  - Paraphrase detection
  - Language-image matching

Our focus: *discriminative* sentence modeling

- Classify a sentence according to a certain criterion
Sentiment analysis

A movie review
An idealistic love story that brings out the latent 15-year-old romantic in everyone.

The sentiment?

Positive Neutral Negative
Feature Engineering

- Bag-of-words
- $n$-gram
- More dedicated ones, e.g., [Silva et al., 2011]...

**Problem:** Sentence modeling is usually NON-TRIVIAL

**Example [Socher et al., 2011]**

white blood cells destroying an infection
an infection destroying white blood cells

Kernel Machines, e.g., SVM

- Circumvent explicit feature representation
- Crucial to design the kernel function, which summarizes all data information
Neural networks

Automatic feature learning

- Word embeddings [Mikolov et al., 2013]
- Paragraph vectors [Le and Mikolov, 2014]

Prevailing neural sentence models

- Convolutional neural networks (CNNs) [Collobert and Weston, 2008]
- Recursive neural networks (RNNs) [Socher et al., 2011]
  - A variant: Recurrent neural networks
**Convolutional Neural Networks (CNNs)**

- Effective feature learning
- Unable to capture tree structural information
“Are tree structures necessary for deep learning of representations?”

Example [Pinker, 1994]

The dog the stick the fire burned beat bit the cat.
If if if it rains it pours I get depressed I should get help.
That that that he left is apparent is clear is obvious.
CNNs versus Sentence Structures

Tree structure

The dog the stick the fire burned beat bit the cat.

Convolution
Recursive Neural Networks (RNNs)

- Structure-sensitive
- Long propagation path

Softmax

Representing hidden layers as vectors recursively

\[ w_l \quad w_r \]

\[ w_l \quad w_r \]
Long Propagation Path

- Burying illuminating information under complicated structure
- Gradient blowup or vanishing
Our Intuition

Can we combine the merits of CNNs and RNNs
- Having short propagation path like CNNs
- Capturing structure info like RNNs

Our solution:

Tree-Based Convolutional Neural Network (TBCNN)
**Outline**

1. Introduction & Related Work

2. Tree-Based Convolution
   - c-TBCNN
   - d-TBCNN

3. Experimental Results
   - Experiment I: Sentiment Analysis
   - Experiment II: Question Classification
   - Model Analysis

4. Conclusion
Architecture of TBCNN

Max pooling by heuristics

Hidden layer  Output layer

Parsing tree of a sentence

Extracted features by tree-based convolution

Lili Mou et al. (Peking University)  TBCNN for Sentence Modeling
Technical Points

How to Represent nodes as vectors in consistency trees?
How to Handle nodes with different numbers of children in dependency trees?
How to Pool over varying sized and shaped structures?
c-TBCNN

- Pretrain an RNN and fix
- Perform convolution

E.g., A convolutional window of depth 2
i.e., a parent \( p \) with children \( l \) and \( r \)

\[
y = f \left( W_p^{(c)} p + W_l^{(c)} c_l + W_r^{(c)} c_r + b^{(c)} \right)
\]
Remark on Complexity

- Exponential to the window depth
- Linear to the number of nodes

☑ Tree-based convolution does not add to complexity,
☐ But is less flexible than “flat” CNNs.
Associate weights with dependency types (e.g., nsubj, dobj) rather than positions

\[ y = f \left( W_p^{(d)} p + \sum_{i=1}^{n} W_{r[c_i]}^{(d)} c_i + b^{(d)} \right) \]

\( r[c_i] \): relation of between \( p \) and \( c_i \)
Pooling Heuristics

- Global pooling
- 3-slot pooling for c-TBCNN
- $k$-slot pooling for d-TBCNN
Sentiment Analysis

Dataset

- Stanford sentiment tree bank
- 5 labels: ++ / + / 0 / - / --
- 8544/1101/2210 sentences, ~150k phrases

Our settings

- 5-way classification + binary classification
- Training: sentences + phrases
- Testing: sentences only

<table>
<thead>
<tr>
<th>Data samples</th>
<th>Label</th>
</tr>
</thead>
<tbody>
<tr>
<td>Offers that rare combination of entertainment and education.</td>
<td>++</td>
</tr>
<tr>
<td>An idealistic love story that brings out the latent 15-year-old romantic in everyone.</td>
<td>+</td>
</tr>
<tr>
<td>Its mysteries are transparently obvious, and it’s too slowly paced to be a thriller.</td>
<td>--</td>
</tr>
<tr>
<td>Group</td>
<td>Method</td>
</tr>
<tr>
<td>------------</td>
<td>---------------------------------</td>
</tr>
<tr>
<td>Baseline</td>
<td>SVM</td>
</tr>
<tr>
<td></td>
<td>Naïve Bayes</td>
</tr>
<tr>
<td>CNNs</td>
<td>1-layer convolution</td>
</tr>
<tr>
<td></td>
<td>Deep CNN</td>
</tr>
<tr>
<td></td>
<td>Non-static</td>
</tr>
<tr>
<td></td>
<td>Multichannel</td>
</tr>
<tr>
<td>RNNs</td>
<td>Basic</td>
</tr>
<tr>
<td></td>
<td>Matrix-vector</td>
</tr>
<tr>
<td></td>
<td>Tensor</td>
</tr>
<tr>
<td></td>
<td>Tree LSTM</td>
</tr>
<tr>
<td></td>
<td>Deep RNN</td>
</tr>
<tr>
<td>Recurrent</td>
<td>LSTM</td>
</tr>
<tr>
<td></td>
<td>bi-LSTM</td>
</tr>
<tr>
<td>Vector</td>
<td>Word vector avg.</td>
</tr>
<tr>
<td></td>
<td>Paragraph vector vector</td>
</tr>
<tr>
<td>TBCNNs</td>
<td>c-TBCCNN</td>
</tr>
<tr>
<td></td>
<td>d-TBCCNN</td>
</tr>
<tr>
<td>Group</td>
<td>Method</td>
</tr>
<tr>
<td>------------</td>
<td>-----------------------------</td>
</tr>
<tr>
<td>Baseline</td>
<td>SVM</td>
</tr>
<tr>
<td></td>
<td>Naïve Bayes</td>
</tr>
<tr>
<td>CNNs</td>
<td>1-layer convolution</td>
</tr>
<tr>
<td></td>
<td>Deep CNN</td>
</tr>
<tr>
<td></td>
<td>Non-static</td>
</tr>
<tr>
<td></td>
<td>Multichannel</td>
</tr>
<tr>
<td>RNNs</td>
<td>Basic</td>
</tr>
<tr>
<td></td>
<td>Matrix-vector</td>
</tr>
<tr>
<td></td>
<td>Tensor</td>
</tr>
<tr>
<td></td>
<td>Tree LSTM</td>
</tr>
<tr>
<td></td>
<td>Deep RNN</td>
</tr>
<tr>
<td>Recurrent</td>
<td>LSTM</td>
</tr>
<tr>
<td></td>
<td>bi-LSTM</td>
</tr>
<tr>
<td>Vector</td>
<td>Word vector avg.</td>
</tr>
<tr>
<td></td>
<td>Paragraph vector</td>
</tr>
<tr>
<td>TBCNNs</td>
<td>c-TBCCNN</td>
</tr>
<tr>
<td></td>
<td>d-TBCNN</td>
</tr>
</tbody>
</table>
Question Classification

Dataset

- 5452 training + 500 test
- Labels
  - abbreviation
  - entity
  - description
  - human
  - location
  - numeric

<table>
<thead>
<tr>
<th>Data samples</th>
<th>Label</th>
</tr>
</thead>
<tbody>
<tr>
<td>What is the temperature at the center of the earth?</td>
<td>number</td>
</tr>
<tr>
<td>What state did the Battle of Bighorn take place in?</td>
<td>location</td>
</tr>
</tbody>
</table>
# Results

<table>
<thead>
<tr>
<th>Method</th>
<th>Acc. (%)</th>
<th>Reported in</th>
</tr>
</thead>
<tbody>
<tr>
<td>SVM 10k features + 60 rules</td>
<td>95.0</td>
<td>[Silva et al., 2011]</td>
</tr>
<tr>
<td>CNN-non-static</td>
<td>93.6</td>
<td>[Kim, 2014]</td>
</tr>
<tr>
<td>CNN-mutlichannel</td>
<td>92.2</td>
<td>[Kim, 2014]</td>
</tr>
<tr>
<td>RNN</td>
<td>90.2</td>
<td>[Zhao et al., 2015]</td>
</tr>
<tr>
<td>Deep-CNN</td>
<td>93.0</td>
<td>[Kalchbrenner et al., 2014]</td>
</tr>
<tr>
<td>Ada-CNN</td>
<td>92.4</td>
<td>[Zhao et al., 2015]</td>
</tr>
<tr>
<td>c-TBCNN</td>
<td>94.8</td>
<td>Our implementation</td>
</tr>
<tr>
<td>d-TBCNN</td>
<td>96.0</td>
<td>Our implementation</td>
</tr>
</tbody>
</table>
### Model Analysis: Pooling Methods

<table>
<thead>
<tr>
<th>Model</th>
<th>Pooling method</th>
<th>5-class accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>c-TBCNN</td>
<td>Global</td>
<td>48.48 ± 0.54</td>
</tr>
<tr>
<td></td>
<td>3-slot</td>
<td>48.69 ± 0.40</td>
</tr>
<tr>
<td>d-TBCNN</td>
<td>Global</td>
<td>49.39 ± 0.24</td>
</tr>
<tr>
<td></td>
<td>2-slot</td>
<td>49.94 ± 0.63</td>
</tr>
</tbody>
</table>

**Remarks**
- Averaged over 5 random initializations
- Hyperparameters predefined, less optimal
Model Analysis: Sentence Length

Reimplemented RNN: 42.7% accuracy, slightly lower than 43.2% reported in [Socher et al., 2011]
“The stunning dreamlike visual will impress even those who have little patience for Euro-film pretension.”
“The stunning dreamlike visual will impress even those who have little patience for Euro-film pretension.”
“The stunning dreamlike visual will impress even those who have little patience for Euro-film pretension.”
Outline

1 Introduction & Related Work

2 Tree-Based Convolution
   - c-TBCNN
   - d-TBCNN

3 Experimental Results
   - Experiment I: Sentiment Analysis
   - Experiment II: Question Classification
   - Model Analysis

4 Conclusion
## Conclusion

<table>
<thead>
<tr>
<th>Structure</th>
<th>Way of information propagation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Flat</td>
<td>Iterative</td>
</tr>
<tr>
<td>Tree</td>
<td>Sliding</td>
</tr>
</tbody>
</table>

- Iterative: Recurrent
- Sliding: Convolution, Tree-based convolution
Thank you for listening!

Q & A
References


A convolutional neural network for modelling sentences.
In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics.

Convolutional neural networks for sentence classification.

Distributed representations of sentences and documents.

Distributed representations of words and phrases and their compositionality.
In Advances in Neural Information Processing Systems.

The Language Instinct: The New Science of Language and Mind.
Penguin Press.
