ELMo: Deep contextualized word representation

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, Luke Zettlemoyer

NAACL-2018
Overview

• A new deep contextualized word representation (ELMo) that:
  • Model complex use of word under specific context
• How:
  • Generate task-specific representations
  • Combine representations in every layers
• ELMo increases the performance of existing models in 6 NLP tasks
• Analysis: ELMo capture abstract meaning on upper layer
Warm up

Deep contextualized word representation

- Hidden representation from all layers
- The representation for each word depends on the entire context
- The fundamental unit
- Embedding

Top layer
All layers

Y=f(word)
Y=f(word, context)

Character
Word
Sentence
Document

One-hot vector
Fix-size distributed vector
One-hot vector

• Working directly with word is extremely hard and inefficient
  • One-hot vector becomes very large
  • Spare representation
  • Lack of operators

Rome = [1, 0, 0, 0, 0, 0, 0, ..., 0]
Paris = [0, 1, 0, 0, 0, 0, 0, ..., 0]
Italy = [0, 0, 1, 0, 0, 0, 0, ..., 0]
France = [0, 0, 0, 1, 0, 0, 0, ..., 0]
Word embedding

- Ideal embedding
  - Dense vector
  - Fix-size vector
  - Context-aware
  - Math-operator compatibility

- Example:
  - Glove (Stanford)
  - Word2vec (Google)
  - fastText (Facebook)

## Context-aware embedding

<table>
<thead>
<tr>
<th>Source</th>
<th>Nearest Neighbors</th>
</tr>
</thead>
<tbody>
<tr>
<td>GloVe</td>
<td>play</td>
</tr>
<tr>
<td></td>
<td>playing, game, games, played, players, plays, player, Play, football, multiplayer</td>
</tr>
<tr>
<td>biLM</td>
<td>Chico Ruiz made a spectacular play on Alusik’s grounder {…}</td>
</tr>
<tr>
<td></td>
<td>Kieffer, the only junior in the group, was commended for his ability to hit in the clutch, as well as his all-round excellent play.</td>
</tr>
<tr>
<td></td>
<td>Olivia De Havilland signed to do a Broadway play for Garson {…}</td>
</tr>
<tr>
<td></td>
<td>{…} they were actors who had been handed fat roles in a successful play, and had talent enough to fill the roles competently, with nice understatement.</td>
</tr>
</tbody>
</table>

Table 4: Nearest neighbors to “play” using GloVe and the context embeddings from a biLM.
BiLM

• ELMo: Embedding from Language Models
• Embeddings are built from bidirectional language model (biLM)
  • Forward:
    \[ p(t_1, t_2, \ldots, t_N) = \prod_{k=1}^{N} p(t_k \mid t_1, t_2, \ldots, t_{k-1}) \]
  • Backward:
    \[ p(t_1, t_2, \ldots, t_N) = \prod_{k=1}^{N} p(t_k \mid t_{k+1}, t_{k+2}, \ldots, t_N) \]
BiLM(2)

- Bidirectional Language Model
  - Predict next word in bi-direction with LSTM

![Diagram of BiLM(2) model with example words and hidden states](image)
**Method (3)**

- Gather representations on all layers

\[ R_k = \{ x_k^{LM}, \overrightarrow{h}_{k,j}^{LM}, \overleftarrow{h}_{k,j}^{LM} \mid j = 1, \ldots, L \} = \{ h_{k,j}^{LM} \mid j = 0, \ldots, L \}, \]

- Gather representations from different task

\[ \text{ELMo}_{k}^{task} = E(R_k; \Theta^{task}) = \gamma^{task} \sum_{j=0}^{L} s_j^{task} h_{k,j}^{LM}. \]
Integration

Enhance inputs with ELMos

Corpus

Train

biLMs

Usual inputs

have

a

nice

https://www.slideshare.net/shuntaroy/a-review-of-deep-contextualized-word-representations-peters-2018
### Experiment

<table>
<thead>
<tr>
<th>TASK</th>
<th>PREVIOUS SOTA</th>
<th>OUR BASELINE</th>
<th>ELMo + BASELINE</th>
<th>INCREASE (ABSOLUTE/RELATIVE)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q&amp;A</td>
<td>SQuAD</td>
<td>Liu et al. (2017)</td>
<td>84.4</td>
<td>81.1</td>
</tr>
<tr>
<td>SNLI</td>
<td>Chen et al. (2017)</td>
<td>He et al. (2017)</td>
<td>81.7</td>
<td>84.6</td>
</tr>
<tr>
<td>SRL</td>
<td></td>
<td></td>
<td>88.0</td>
<td>88.7 ± 0.17</td>
</tr>
<tr>
<td>Coref</td>
<td>Lee et al. (2017)</td>
<td>67.2</td>
<td>70.4</td>
<td>3.2 / 9.8%</td>
</tr>
<tr>
<td>NER</td>
<td>Peters et al. (2017)</td>
<td>91.93 ± 0.19</td>
<td>90.15</td>
<td>92.22 ± 0.10</td>
</tr>
<tr>
<td>SST-5</td>
<td>McCann et al. (2017)</td>
<td>53.7</td>
<td>51.4</td>
<td>54.7 ± 0.5</td>
</tr>
</tbody>
</table>

Table 1: Test set comparison of ELMo enhanced neural models with state-of-the-art single model baselines across six benchmark NLP tasks. The performance metric varies across tasks – accuracy for SNLI and SST-5; $F_1$ for SQuAD, SRL and NER; average $F_1$ for Coref. Due to the small test sizes for NER and SST-5, we report the mean and standard deviation across five runs with different random seeds. The “increase” column lists both the absolute and relative improvements over our baseline.
## Analysis (1)

<table>
<thead>
<tr>
<th>Model</th>
<th>F$_1$</th>
<th>Model</th>
<th>Acc.</th>
</tr>
</thead>
<tbody>
<tr>
<td>WordNet 1st Sense Baseline</td>
<td>65.9</td>
<td>Collobert et al. (2011)</td>
<td>97.3</td>
</tr>
<tr>
<td>Raganato et al. (2017a)</td>
<td>69.9</td>
<td>Ma and Hovy (2016)</td>
<td>97.6</td>
</tr>
<tr>
<td>Iacobacci et al. (2016)</td>
<td><strong>70.1</strong></td>
<td>Ling et al. (2015)</td>
<td><strong>97.8</strong></td>
</tr>
<tr>
<td>CoVe, First Layer</td>
<td>59.4</td>
<td>CoVe, First Layer</td>
<td>93.3</td>
</tr>
<tr>
<td>CoVe, Second Layer</td>
<td>64.7</td>
<td>CoVe, Second Layer</td>
<td>92.8</td>
</tr>
<tr>
<td>biLM, First layer</td>
<td>67.4</td>
<td>biLM, First Layer</td>
<td>97.3</td>
</tr>
<tr>
<td>biLM, Second layer</td>
<td>69.0</td>
<td>biLM, Second Layer</td>
<td>96.8</td>
</tr>
</tbody>
</table>

Table 5: All-words fine grained WSD F$_1$. For CoVe and the biLM, we report scores for both the first and second layer biLSTMs.

Table 6: Test set POS tagging accuracies for PTB. For CoVe and the biLM, we report scores for both the first and second layer biLSTMs.
Analysis (2)

Most models preferred “syntactic (probably)” features

Even in sentiment analysis

Figure 2: Visualization of softmax normalized biLM layer weights across tasks and ELMo locations. Normalized weights less then 1/3 are hatched with horizontal lines and those greater then 2/3 are speckled.
Figure 1: Comparison of baseline vs. ELMo performance for SNLI and SRL as the training set size is varied from 0.1% to 100%.
Long Short-term Memory (LSTM)

http://colah.github.io/posts/2015-08-Understanding-LSTMs/
Q & A