Attention and CNN empowered BiDAF for SQuAD

Haojun Li
Department of Computer Science, Stanford University

Abstract
The SQuAD dataset introduced in [1] formulate the Question and Answer problem as a span prediction problem with adversarially written unanswerable questions. In this project I aim to

- Establish a better baseline by adding character-level embedding,
- Creatively use Encoder Block inspired from [2]
- Experiment with a multi-headed BiDAF attention layer

I have shown that by combining CNN-based and RNN-based encoders, the model performs better than those encoders alone, and significantly better than the baseline model. On the other hand multi-headed BiDAF attention models perform worse due to overfitting and other fundamental model restrictions.

Data
The SQuAD dataset consists of entries of a context paragraph, a question, and an answer, where the answer can be found as an exact match to a span of words in the context. An example is shown below:

- Question: What president eliminated the Christian position in the curriculum?
- Context: Charles W. Eliot, president 1869–1909, eliminated the favored position of Christianity from the curriculum while opening it to student self-direction. While Eliot was the most crucial figure in the secularization of American higher education, he was motivated not by a desire to secularize education, but by Transcendentalist Unitarian convictions. Derived from William Ellery Channing and Ralph Waldo Emerson, these convictions were focused on the dignity and worth of human nature, the rights and ability of each person to perceive truth, and the indwelling God in each person.
- Answer: Charles W. Eliot

Results & Analysis

<table>
<thead>
<tr>
<th>Model</th>
<th>Note</th>
<th>EM</th>
<th>F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>BiDAF Baseline</td>
<td></td>
<td>55</td>
<td>58</td>
</tr>
<tr>
<td>BiDAF-char</td>
<td>new baseline</td>
<td>61.18</td>
<td>64.2</td>
</tr>
<tr>
<td>Multi-BiDAF feedforward</td>
<td></td>
<td>58.33</td>
<td>61.14</td>
</tr>
<tr>
<td>Multi-BiDAF max pool</td>
<td></td>
<td>60.46</td>
<td>63.59</td>
</tr>
<tr>
<td>EB only</td>
<td></td>
<td>55.96</td>
<td>59.71</td>
</tr>
<tr>
<td>EB + RNN fine-tune</td>
<td></td>
<td>64.79</td>
<td>67.87</td>
</tr>
<tr>
<td>EB + RNN Test split results</td>
<td></td>
<td>62.671</td>
<td>66.045</td>
</tr>
</tbody>
</table>

Table 1: Results from experiments

Multi-headed BiDAF Attention

I added multiple attention heads to the BiDAF attention layer:

\[ S^{(k)}_{ij} = \text{softmax}(S^{(k)}_{ij}) \]

Then, I calculate multiple attention outputs

\[ S^{(k)}_i = \text{softmax}(S^{(k)}_{ij}) \]

Combining them in 2 ways

- Taking max: \( a_{ij} = \max_k (S^{(k)}_{ij}), q_{ij} = \max_k (q^{(k)}_{ij}) \), \( a_{ij} = \max_k (a^{(k)}_{ij}), b_{ij} = \max_k (b^{(k)}_{ij}) \).
- Linear Feedforward:

\[ a_i = W_c[a^{(1)}_1, a^{(2)}_2, ..., a^{(k)}_k] + \text{bias}_a \]

Conclusion

- Basic improvements such as character embedding and hyperparameter tuning significantly improves the baseline.
- Incorporating an Encoder Block to the Embedding Encoding layer inspired by QANet[2] is able to greatly improve the new baseline.
- Multi-headed BiDAF performs worse due to overfitting and fundamental model restrictions.

Future Works

A lot still need to be done if I have more time

- Use L2 regularization to solve overfitting
- Added self-attention after BiDAF as shown in [4]
- Use Transformer-XL as shown in [5] to improve long term dependency recognition

References