Challenges:
Recurrency-Free Question Answering with Unanswerable Questions
Michael Hazard, Ian Hodge, Daniel Semeniuta
{cmhazard, ihodge, dsemniu}@stanford.edu

Introduction

Goal: Build a system for question answering as well as identify unanswerable questions in the SQuAD 2.0 dataset.

Question answering: Identifying a sequence in a context paragraph that correctly answers an input query.

Approach

Evaluation Metric: F1 and EM scores on the Dev and Test set.

Data: SQuAD 2.0, a dataset of over 150,000 questions, both answerable and unanswerable.

Word Embeddings: Stanford’s pretrained GLoVE embeddings.

Experimental Details:
- Adam Optimizer with learning rate warmup and weight decay.
- Character-level embedding ablation.
- Gradient accumulation for larger effective batch sizes.
- Variation in number of heads, hidden size, encoder repetition.
- Stochastic Depth Dropout.
- Weight initialization ablation testing.

Results

Table: Dev F1 scores:

<table>
<thead>
<tr>
<th>Model</th>
<th>F1 Score</th>
<th>EM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline BiDAF</td>
<td>58</td>
<td>59</td>
</tr>
<tr>
<td>Tuned BiDAF</td>
<td>62.814</td>
<td>59.267</td>
</tr>
<tr>
<td>Tuned BiDAF + Char Embeddings</td>
<td>65.731</td>
<td>62.225</td>
</tr>
<tr>
<td>QANet</td>
<td>55.64</td>
<td>52.26</td>
</tr>
<tr>
<td>QANet + Char Embeddings</td>
<td>56.91</td>
<td>53.18</td>
</tr>
</tbody>
</table>

Model

QANet Model:
- Embedding Layer
- Embedding Encoder
- Context-Query Attention Layer
- Model Encoder Layer
- Output Layer

Answer Verifier:
- Embedding Layer
- Transformer Layer
- Output Layer
- Option to pre-train as a language model

Error Analysis

Misidentified Specifier:
Question: What sort of motion did Newcomen’s steam engine continuously produce?
Context: In 1781 James Watt patented a steam engine that produced continuous rotary motion.
Prediction: rotary

Incorrect Answer Length:
Question: What is the Chinese name for the Yuan dynasty?
Context: The Yuan dynasty (Chinese: 元朝; pinyin: Yuán Cháo), officially the Great Yuan (Chinese: 大元; pinyin: Dà Yuán; Mongolian: Yehe Yuan Ulus[a])...
Prediction: 元朝; pinyin: Yuán Cháo), officially the Great Yuan

Challenges/Conclusions

- The model was too large to fit onto any of the GPUs provided with a reasonable batch size. As a result, training took much longer than anticipated. This defeated the purported speed gains of the non-sequential architecture.
- The long training cycle (~12 hours) made debugging and tuning very difficult. We often had to try multiple changes at once which made it difficult to narrow down our problems.
- We found the transformer architecture to be extremely fragile and small tunes to the hyper parameters made drastic positive and negative changes.
- Future work would include incorporating the verifier with our BiDAF and QANet models, with a possible ensemble approach.

References Cited