Introduction

Question answering is a problem that we have approached from a novel perspective, using word embeddings as the extent of our feature-engineering. While this has produced mediocre results, applying additional feature engineering can improve upon this baseline. Some well-motivated changes produced improved results, such as an exact match feature, while others actually harmed the performance compared to the baseline. A combination of exact match and a lemma match feature. My findings indicate that enhanced feature engineering is useful for fine-tuning a model's performance on a task, but it will not produce significantly improved accuracy.

Background: SQuAD

This project makes use of the official SQuAD 2.0 dataset. The data is split into a train, dev, and test set with each containing 12,924, examples, 4,078 examples, and 3915 examples, respectively.

Question: When B and T cells begin to replicate, what do some of their offspring cells become?

Context: When B cells and T cells are activated and begin to replicate, some of their offspring become long-lived memory cells. Throughout the lifetime of an animal, these memory cells remember each specific pathogen encountered and can react with a strong response if the pathogen is detected again. This is adaptive because it occurs during the lifetime of an individual as an adaptation to infection with that pathogen and prepares the immune system for future challenges. Immunological memory can be in the form of either passive short-term memory or active long-term memory.

Answer: long-lived memory cells

Background: BIDAF

Baseline Model

The changes proposed in the project are built on top of a standard BIDAF model for Question Answering on the SQuAD dataset. The BIDAF model is composed of an Embedding layer, an RNN encoding layer, a BIDAF attention layer, another RNN encoding layer, and finally the BIDAF output layer. We used a hidden size of 100 and a learning rate of 0.1. We trained for 30 epochs on the dev set.

Feature Engineering

BIDAF achieves decent results on QA tasks using 300-dimensional GloVe embeddings. Such straightforward input to vector approaches for deep learning networks is very common. The intuition for such simplistic approaches is that the deep learning networks should be able to learn more complex syntactic and semantic representations within the networks. In some cases, however, our findings indicate that the exact match feature, while not always beneficial, can improve performance compared to the baseline. A combination of exact match and a lemma match feature. My findings indicate that enhanced feature engineering is useful for fine-tuning a model's performance on a task, but it will not produce significantly improved accuracy.

Results

The BIDAF model that used 300 dimension GloVe embeddings was trained for 30 epochs on the training dataset. The results for the baseline model and the baseline augmented with the feature engineering on the dev set are reported in Table 1 below. The primary metric is the F1 score, the harmonic mean of precision and recall. Mathematically, it is 2 prediction recall / (prediction + recall). We also have an Exact Match (EM) score to indicate the percentage of exactly correct answers.

<table>
<thead>
<tr>
<th>EM</th>
<th>F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline</td>
<td>57.62</td>
</tr>
<tr>
<td>Exact Match feature</td>
<td>60.41</td>
</tr>
<tr>
<td>Lemma Match feature</td>
<td>59.44</td>
</tr>
<tr>
<td>Exact Match and Lemma Match feature</td>
<td>56.84</td>
</tr>
<tr>
<td>Part of Speech (POS)</td>
<td>58.36</td>
</tr>
<tr>
<td>Part of Speech (POS) + Embedding</td>
<td>56.75</td>
</tr>
</tbody>
</table>

Table 1: Results for baseline and feature augmented models on the dev set

These results are quite fascinating, because I expected more complicated features to produce more complex representations, such as the part of speech embeddings. However, this was not the case. Even the two highest performing features, the exact match and lemma match, did worse when combined. The addition of these two features did not contribute significantly more than was lost by decreasing the representation of the word embeddings to accommodate them. I am somewhat surprised by the worse performance of the exact match feature on the test set versus the dev set, as it appeared to be a significant improvement over the baseline model. However, it appears that the training gains on the dev set did not generalize well.

Analysis

For Figure 1, the baseline model is in orange, the exact match is in dark blue, the lemma match is cyan, and the exact/lemma match is pink. We can see here that the augmented models achieved similar dev set performance compared to the baseline model's best performance in about a third of the time. I believe that the reason these models achieve similar performance to much faster is that the baseline model takes millions of iterations to encode the exact relationships through attention that we augmented the model with.

Conclusion

A useful target for feature engineering in the context of NLP and attention is something that captures a structural regularity in the data that is not already explicit. The value of augmenting your model with such a feature is the potential to reduce the training requirements to achieve otherwise similar performance. However, the beauty of deep learning is that it is capable of learning the well-defined patterns in data without needing to be told, if it occurs eventually.

References


Figure 1: Training progress of various augmented models compared to the baseline.