Is Feature Engineering Futile?

Jacob A. Smith [1]

[1]Computer Science, Stanford

Introduction

Question answering is a problem that we have approached from a naive 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 129,941 examples, 6078 examples, and 5915 examples respectively.

Question: When B cells 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 mount 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

Question: After 1945, what challenged the British empire?

Context: In World War II, Charles de Gaulle and the Free French used the overseas colonies as bases from which they fought to liberate France. However after 1945 anti-colonial movements began to challenge the Empire. France fought and lost a bitter war in Vietnam in the 1950s. Whereas they won the war in Algeria, the French leader at the time, Charles de Gaulle, decided to grant Algeria independence anyway in 1962. Its settlers and many local supporters relocated to France. Nearly all of France's colonies gained independence by 1960, but France retained great financial and diplomatic influence. It has repeatedly sent troops to assist its former colonies in Africa in suppressing insurrections and coups d'état.

Answer: N/A

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 [2]. We used a hidden size of 100 and a drop probability of 0.2. We used a batch size of 64, an ema decay of 0.999, and a learning rate of 0.5. We trained for 30 epochs.

Feature Engineering

BiDAF achieves decent results on QA tasks using just 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 network. In that case, my question is, what if we jump start that process by passing in information that a human answering similar questions might be using to come up with a response. For example, a human might highlight words in the context that appear in the question. Here are the features that I engineered inspired by Chen et. al. [1]

1. **Exact Match** Binary indicator for each context word if it appears in the question.
2. **Lemma Match**. Binary indicator for each context word if its lemma form appears in the lemmatized question.
3. **Exact Match and Lemma Match** Two binary indicators for each of the above.
4. **Part of Speech** An encoding of the part of speech for each context word.

Results

The naive 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 engineered features on the dev set are reported in Table 1 below.

The primary metric of evaluation is F1, the harmonic mean of precision and recall. Mathematically, it is 2 x prediction x recall / (precision + recall). We also have an Exact Match (EM) score to indicate the percentage of exactly correct answers.

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

	EM	F1
Baseline	57.64	61.26
Exact Match feature	**60.41**	**63.7**
Lemma Match feature	59.44	62.49
Exact Match and Lemma Match feature	58.46	61.84
Part of Speech One Hot	56.34	59.75
Part of Speech Embedding (6)	58.36	61.67
Part of Speech Embedding (10)	56.75	60.22

Table 2. Results for baseline and feature augmented models on the test set

	EM	F1
Baseline	57.524	61.01
Exact Match feature	57.63	60.90

These results are quite fascinating, because I expected more complicated features to produce more useful 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 realized gains on the dev set did not generalize well.

Analysis

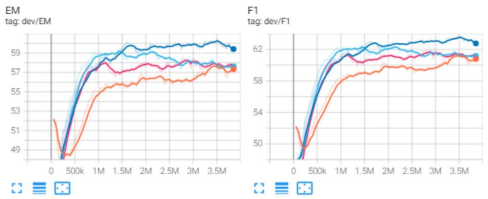
Drawbacks

We have seen that this feature engineering has not provided significant benefits in terms of accuracy compared to the baseline model. The feature engineering is also less straightforward to implement than just using embeddings.

Benefits

However, there are some other benefits that may have a real world value. The most significant of these was the rate at which the exact match, lemma match, and exact/lemma match augmented models learned. Compared to the baseline, which first took a dip before improving in F1 and EM, these augmented models began improving immediately and performed better more rapidly.

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



For figure 1, the baseline model is in orange, the exact match is dark blue, the lemma match is cyan, and the exact/lemma match is pink. We can see here that these augmented models achieved similar dev set performance compared with the baseline model's best performance in about a third of the time. I believe that the reason these models achieve similar performance so 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 so, if only eventually.

References

[1] Danqi Chen, Adam Fish, Jason Weston, and Antoine Bordes.
Reading wikipedia to answer open-domain questions.
In *arXiv preprint arXiv:1704.00051*, 2017.

[2] Minjoon Seo, Aniruddha Kembhavi, Ali Farhadi, and Hannaneh Hajishirzi.
Bidirectionalattention flow for machine comprehension.
In *arXiv preprint arXiv:1611.01603*, 2016.

jacobas@stanford.edu