Training computers to understand text has gained significant popularity over the past several years due to the many applications it enables. Imagine being able to ask a computer to better understand a piece of text. Significant research has been done to design models to compete in this difficult challenge. In general, submissions fall into one of two categories: those that use pre-trained contextual embeddings (PCE) such as ELMo and BERT, and those that do not. PCE models tend to have much higher performance than non-PCE models, however come with an added level of complexity. We chose to focus our project on enhancing existing non-PCE models.

For the project we used the SQuAD 2.0 dataset, one of the most popular reading comprehension benchmarks. The dataset comprises many questions, each associated with a context paragraph that may or may not include the answer. The goal of the project is to train a computer to answer the questions as correctly as possible—providing a measure for how well the computer can ‘understand’ text.

The model using full embeddings (word, character, POS, NER tag, and TF) with tuned hyperparameters performed the best out of the 7 models tested. Compared to the baseline model, this is an EM score increase of 3.66 and an F1 score increase of 3.30.

### Ablation Studies
Of all model components analyzed, the character-level embeddings had the greatest effect. It is likely that, with the addition of character embedding information, the model is better able to handle instances of unknown words appearing in either the question or context by using character-level information to gain insight into these unknown words.

We also see evidence that the POS embeddings provided a significant positive effect when the other two were considered very little; since performance drops off heavily when removing the POS embeddings, but remains nearly constant when removing both NER and TF.

### Characteristic Examples
1. **Best BIDAF model**
   - Often chose answers close to correct, but too lengthy
   - **Ex:** *Question:* "What is the most important type of Norman art preserved at churches?" *Correct:* 'mosaics’, *Model:* ‘sculptured forts, capitals, and more importantly mosaics’

2. **QANet model observed errors**
   - Demonstrated an ability to answer with the correct type of item or idea, but often struggled with choosing which of the items from the context to choose from

### Conclusions/Future Work
BIDAF: Supplementing word-level embeddings with character-level embeddings leads to better performance. However, adding only the part-of-speech feature seems to be nearly as effective as adding all three additional token features. Further tuning the model’s hyperparameters and adding different, more descriptive token feature embeddings could result in even better performance.

QANet: For the QANet model there are many opportunities to further improve its performance. Many more experiments could be run using different hyperparameters. Specifically, decaying the learning rate at later time steps and increasing the batch size and the model’s hidden size would likely improve the sometimes erratic training behavior.

### References
[3] Attention is all you need in pytorch repository: https://github.com/facebookresearch/attention-is-all-you-need-pytorch