NLP Project Report

Karthik Hegde: IMT2018509
Saravan S G : IMT2018521
Jishnu V K : IMT2018033

Aim of the Project: To develop a system that can “read” a text book and and answer questions based on it. The system has to return a phrase or sentence that correctly answers the question. It also has to point out where in the text book it got the answer.

Short description of the final System:

The system can be broadly divided into 2 parts: a Reading Comprehension QA model and an information retrieval system that can aid the QA model by supplying the correct context.

For the information retriaval system, we used the BM25 okapi algorithm.
For the QA model, we used a BERT based model. The model takes a question and a context. It returns a single span in the context as the answer to the question. The model has 2 heads that return the start index and end index of the span. The model was trained on the SQUAD dataset, a standard dataset used for reading comprehension.

To input the textbook to the system, it is copy pasted into a text file. Each paragraph in the text book is treated as a seperate context passage. They are divided in the text file using // as the seperator.

The final notebook has many examples of the system answering questions in the squad dataset eval data. It also has examples of answering questions on a chapter of 11^{th} grade NCERT: “environmental chemistry”.

Exploration and details:

Information Retrieval:

We tried 2 different methods for information retrieval, a word embedding approach and BM25 Okapi.

Word embedding approach:
For the word embedding approach, we used word2vec to represent the words. A document was represented as the mean, row-wise max or row-wise min of the word vectors it contains. The full corpus has to be preprocessed for this.

The preprocessing includes:
1) Contraction expansion
2) Lowercasing
3) removing any non alphabet words
4) tokenize using spacy tokenizer. This also helps in lemmatization of words.

The best match was found using cosine similarities. The testing was done on the eval set of SQUAD dataset. A corpus was created by combining all the context passages in the eval set. Testing was done by running each query in the eval set and finding top-1 match amongst the corpus.

This method had poor performance in top-1 match performance. It faired decently well when we allowed it to select top-5. It seems to be affected because many of the context passages are very similar to each other since there are several context passages about the same topic. The model returns a context from the same topic but not the correct context for the question asked. Since our system relies on a single context passage, this was not good.

BM25 Okapi:
BM25 is an information retrieval method that relies on a probabilistic approach. It traces its origin to the 1980s where it was first used in the London City University's Okapi IR system. It falls under the category of term-frequency, inverse-document-frequency methods. Unlike the word2vec method, it doesn't take into account any context or proximity of the search terms.

Without any preprocessing, the algorithm was able to return the correct context 52% of the time and top-5 score was 71%. After experimenting with lemmatization and preprocessing, top-1 score was maximized at 78% and top-5 score was 89%. We settled for this algorithm since it gave us satisfactory results.

Question answering:

Single span model:

We used a BERT based model to create a question answering model. The model architecture used a BERT encoder as the first layer. The query and context are tokenized and input together to the BERT encoder seperated by <SEP> token. 2 parallel single layer feed forward networks were attached to the output of the BERT encoder. These were made to learn the index of the start and the end index of the answer in the context. So, the model learns to extract a span from the context to answer the question. The SQUAD training data has over 80000 example. So training was done for only 1-3 epochs. This gave us a validation accuracy of 71%.

Exploration into multiple spans and reasoning:

We also tried to explore into adding additional reasoning abilities into the QA model. For this we used the Quoref dataset. This dataset contains questions that require additional reasoning and also the ability to extract multiple spans from the context. It is also significantly smaller than SQUAD with only around 20000 examples. Further, many contexts are more than 512 words long which reduces the training data that our model can use to only around 13000 examples.

We first tried to train the existing model. We trained it for a maximum of 50 epochs though most of the EM score increase was seen in the first 5 epochs. The model was able to attain an accuracy of 47%.

We then tried to modify the model. Instead of 2 heads giving the start and end index, we added a single feed forward network that tags each word with “I” or “O”. I indicating that the word is in the answer and O indicating it was outside the answer. The motive for this was to give the model the ability to select more than 1 continuous span for the answer. However, we faced lots of problems with this model. Training did not go well and the model was not learning in an expected way. It seemed to get stuck at non optimal points and not learn anything useful. Lots of time was spent to debug this by training on SQUAD, training on a small subset etc but we could not find the root cause of the problem.