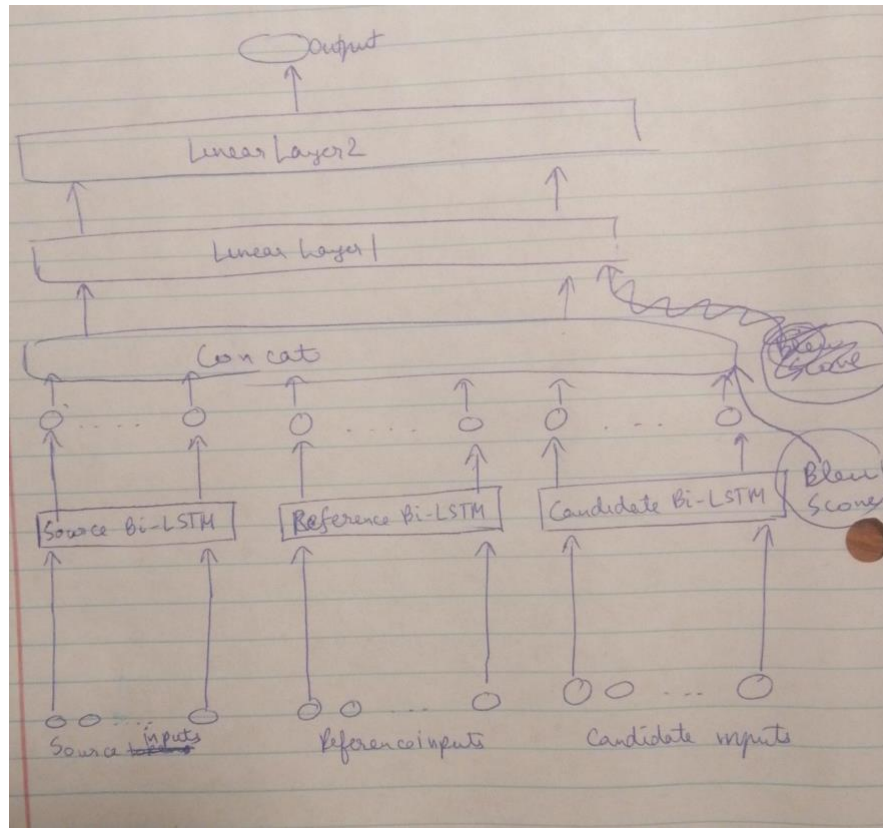


Research Quick Task Report

Model architecture



Motivation for model architecture.

1. Input representation – For each of source, reference and candidate inputs, the input token was formed by the Word Embedding of the word, the NER tag of the word and the POS tag of the word since I wanted the model to recognize the essence of the source, reference and candidate inputs to better draw similarities between them. Specifically, for any word w , the input token was $[WE(w); NER(w); POS(w)]$.
2. I used 3 Bi-LSTMs, one for each of source inputs, reference inputs and candidate inputs gain information about the sequential nature of the source, reference and candidate inputs.
3. I concatenated the Source LSTM outputs, Reference LSTM Outputs, Candidate LSTM Outputs and the Bleu Score and passed them into a two-layer FFNN (two linear layers with ReLU activation) to receive the output – a number between 0 and 1 explaining whether it was closer to a human or machine candidate respectively. I used the FFNN to draw non-linear similarities between the source, reference and candidate LSTM outputs.

Results Analysis.

1. My model predicted 0 (Human) for all the test data (achieving 0.346 average f1 score)

2. The feature selection – word embeddings, POS tags and NER tags were causing my model to overfit the data. This problem occurred primarily due to data sparsity. My model was too big (had too many layers) for it to learn anything.

Potential improvements

1. In a research situation and given a larger time period, I would either accumulate more data which would allow the model to learn more about the task from the word embeddings or I would change the feature representation of each input – experimenting with removing word embeddings, POS and NER tags altogether and using similarity-based measures such as the proportion of words in the reference present in the candidate.
2. I would also improve the model architecture, perhaps incorporating methods similar to NLP attention to draw similarities between the source, candidate and reference sentences instead of just concatenating them and passing them into a FFNN.
3. In experimenting with different models, I would create a train/validation split and plot train and validation loss to determine if I'm overfitting or underfitting the dataset.
4. I would spend more time on hyperparameter tuning. Some hyperparameters to consider – hidden size of LSTMs, FFNNs etc. I would also try to reduce the FFNN input size by making the FFNN input multi-dimensional.