Documentation:

1) The given code has been heavily inspired from a github repo found.

2)My attempt was to understand each step and make modifications to improve the code.

3) Dropout layers at multiple places were made less heavy/removed and the accuracy was checked, and put in a way to optimize the most.

4)For inner product between the embeddings of sentences and questions, cosine similarity was found to give better result as compare to simple dot product.

5) The output dimension of encoding layers was increased in attempt to make better differentiation/matching between sentences/words.

Following is a list of my learnings:

1) Understood how after embedding, the weights pertaining to sentence are calculated.

2) Understood how softmax helps in conversion of sentences to vectors in continuous space.

3) Understood how encoding followed by dropout works, to help with overfitting, yet found it not so useful.

4)Understood how, while embedding the vocab_size helps in maintaining uniformity which helps in finding the vector similarity in embedded space.

5)Application of LSTM layer to implement RNN.

Please replace the test data file with the appropriate one. The paths have been given assuming that the train and test data are in the same directory as the .ipynb file.

The final validation accuracy was approximately 0.82.