Deep Learning Project 3
Natural Language Inference

Problem Statement-
Given two sentences hypothesis and premise, we want to predict whether these pairs of statements are true (entailment), false (contradiction), or undetermined (neutral). The data used here is SNLI (Stanford Natural Language Inference) dataset which has 60000 train examples and 10000 test examples.

Preprocessing-
The hypothesis and premises are pre-processed separately.
Following pre-processing methods are used

1. Text to lowercase
2. Remove punctuation
3. Remove extra white-spaces
4. Lemmatize word- converts words to its base or dictionary form
5. Drop Stop words- Drop the words which make lesser or no sense in sentiment analysis like and, this, the, etc.

Before preprocessing- A woman with a rolling luggage waits on a sidewalk .
After preprocessing- woman roll luggage wait sidewalk

Models-

1. Logistic regression using TF-IDF vectors

The first model is a simple Logistic regression model where TF-IDF features are used. TF-IDF (Term Frequency - inverse document frequency) features are used to vectorize the words. Term frequency measures the frequency of words in a document. Document frequency measures the number of documents in which a given word appear. Inverse document frequency is inverse of the document frequency which is low for a word if it appears in many documents and high for a word that appears rarely.

$$tf\text{-}idf(t, d) = tf(t, d) * \log(N/(df + 1))$$

t is a term
d is a document
df if document frequency
N is the number of documents in corpus

Model Architecture-
After preprocessing, data is converted to tf-idf vectors. These tf-idf vectors are used in training the logistic regression model. Using this model I was getting only 43.75% accuracy. Any changes in the model accuracy were not making much change in the training accuracy.

2. Deep learning models(RNN and LSTM)

After preprocessing, words in each sentence are replaced by a number, and padding is applied. Zero is padded in the required sentences to make the length of each data point the same. After padding the training data is represented is converted to vectors using the pre-trained glove vector model

GloVe-

GloVe is an unsupervised training algorithm for obtaining vector representation for words. It follows the intuition that the vectors corresponding to two similar words lie closer to each other in vector space. Pretrained Glove models are available on nlp.stanford.edu. In this project, I used a 50-dimensional pre-trained GloVe model.

LSTM

To get the model architecture, different values are done on the hyperparameters are tried which is mentioned in the table below.

I also tried Bidirectional LSTM which allows the network to have both forward and backward information about the sequence at any time. It can be seen from the experiments that of all the models BiLSTM works best. These accuracies are recorded at no. of epochs =3. Then finally BiLSTM is trained for 10 epochs

	Train accuracy	**Train loss**
Simple RNN (no. of cells=60)	76.96 %	0.4465
LSTM (no of cells=10)	75.88%	0.4603
LSTM (no. of cells =60) Without preprocessing	78.09%	0.4313
LSTM no. of cells=60 With preprocessing	78.22%	0.4287
Bidirectional LSTM	78.86%	0.4199

Test accuracy

BiLSTM (with text preprocessing , no. of cells=60)	74.13%
Logistic regression with TF-IDF features	46.89%

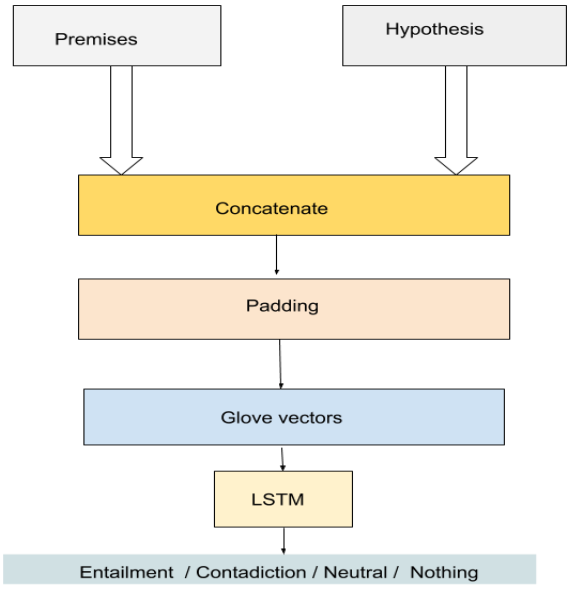

Fig 1: The figure shows deep learning model that was used

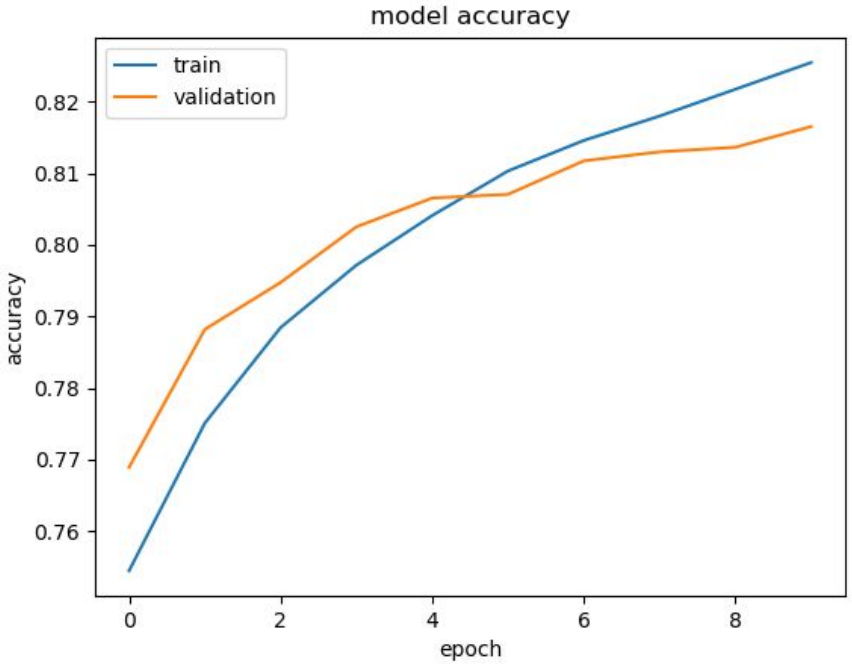

Fig2: training and validation accuracy for deep-learning(LSTM) model

It can be seen that after 10 epochs validation accuracy has almost saturated. I believe still there is a scope of increasing train and validation accuracy by training for more epochs but as the training is time-consuming, the final model was also trained for 10 epochs.