Assignment – 4

Task: To apply RNNs to text and sequence data.

Conditions:

The following conditions need to be applied to the code and re-run to see the performance.

Re-run the example modifying the following:

1) Cutoff reviews after 150 words

2) Restrict training samples to 100

3) Validate on 10,000 samples

4) Consider only the top 10,000 words

5) Before the layers. Bidirectional layer, consider a) an embedding layer, and b) a pretrained word embedding.

Here, we use the IMDB data in our dataset using keras sample dataset. Also if we look at the

Basic design

```
Model: "model_16"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 input_17 (InputLayer)       [(None, None)]            0

 tf.one_hot_4 (TFOpLambda)   (None, None, 10000)       0

 bidirectional_16 (Bidirecti  (None, 64)               2568448
 onal)

 dropout_16 (Dropout)        (None, 64)                0

 dense_16 (Dense)            (None, 1)                 65

=================================================================
Total params: 2,568,513
Trainable params: 2,568,513
Non-trainable params: 0
_________________________________________________________________
```

Using Pre-Trained Word Embeddings:

About **Glove** (info from web):

GloVe (Global Vectors for Word Representation) is a popular algorithm for generating word embeddings, which are numerical representations of words in a high-dimensional space. Here are some highlights about GloVe:

- GloVe is based on the idea that words that appear in similar contexts have similar meanings. It uses co-occurrence statistics from a large corpus of text to learn these contextual relationships between words.
- Unlike other word embedding algorithms, such as word2vec, GloVe directly optimizes a global objective function that captures the overall co-occurrence statistics of the corpus.
- GloVe has been shown to outperform other word embedding algorithms on a variety of natural language processing (NLP) tasks, including word similarity, named entity recognition, and sentiment analysis.
- The pre-trained GloVe embeddings are available in various sizes, from 50-dimensional to 300-dimensional vectors, and can be easily integrated into NLP models.
- GloVe is a popular choice for researchers and developers in the NLP community due to its simplicity, effectiveness, and ease of use.

The following are the performance results using the different techniques in the code.

Solution:

Conditions	First Basic Sequence Model	Embedding Layer	Using Padding & Masking	Using Pretrained word embedding
1.Cutoff reviews after 150 words. 2.Restrict training samples to 100. 3. Validate 10,000 samples. 4. Consider only the top 10,000 words. 5. Before Embedding layer, add a & b and b) a pretrained word embedding	81.1	79.8	80.3	77.8
1.Cutoff reviews after 150 words. 2.Restrict training samples to 800. 3. Validate 10,000 samples. 4. Consider only the top 10,000 words. 5. Before Embedding layer, add a & b and b) a	84.9	83.5	83.5	84.0

pretrained word embedding				
1.Cutoff reviews after 150 words. 2.Restrict training samples to 1600. 3. Validate 10,000 samples. 4. Consider only the top 10,000 words. 5. Before Embedding layer, add a & b and b) a pretrained word embedding	84.6	84.0	83.5	83.6
1.Cutoff reviews after 150 words. 2.Restrict training samples to 2400. 3. Validate 10,000 samples. 4. Consider only the top 10,000 words. 5. Before Embedding layer, add a & b and b) a pretrained word embedding	85.0	83.80	83.4	83.80
1.Cutoff reviews after 150 words. 2.Restrict training samples to 3200. 3. Validate 10,000 samples. 4. Consider only the top 10,000 words. 5. Before Embedding layer, add a & b and b) a pretrained word embedding	84.7	83.7	84.2	83.20
1.Cutoff reviews after 150 words. 2. Restrict training samples to 4000 3. Validate 10,000 samples.	84.2	83.70	83.90	83.6

4. Consider only the top 10,000 words. 5. Before Embedding layer, add a & b and b) a pretrained word embedding				
With_all_training_Samples	88.4	85.90	87.50	87.30

Conclusions:

The above results show the different variations in the training samples and their results. Here we used different techniques to improve the performance of the model, however it's almost equal with the pretrained embedded word. If we observe the performance results, basic sequence network is slightly better in performance.