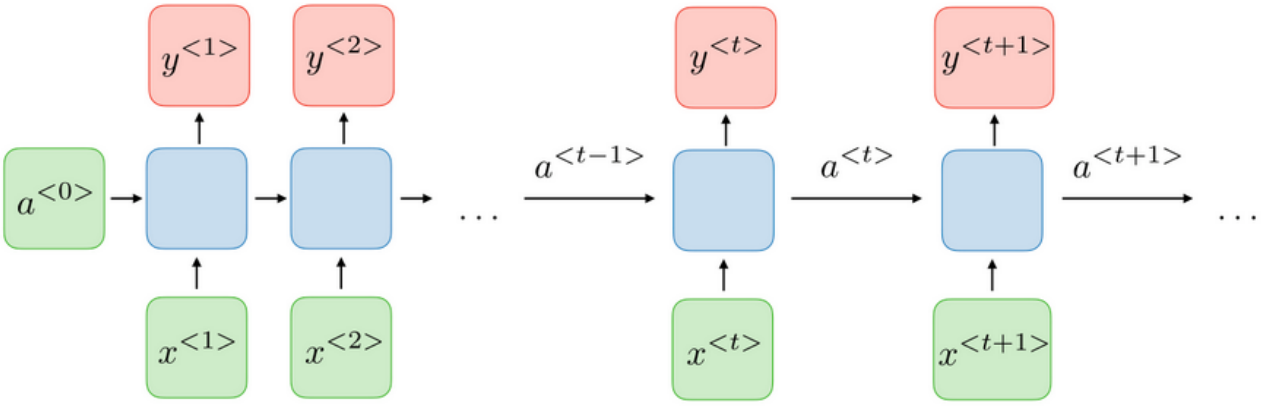
ECE 6524: Deep Learning HW 4

Pranav Chimote

April 20, 2020

Introduction

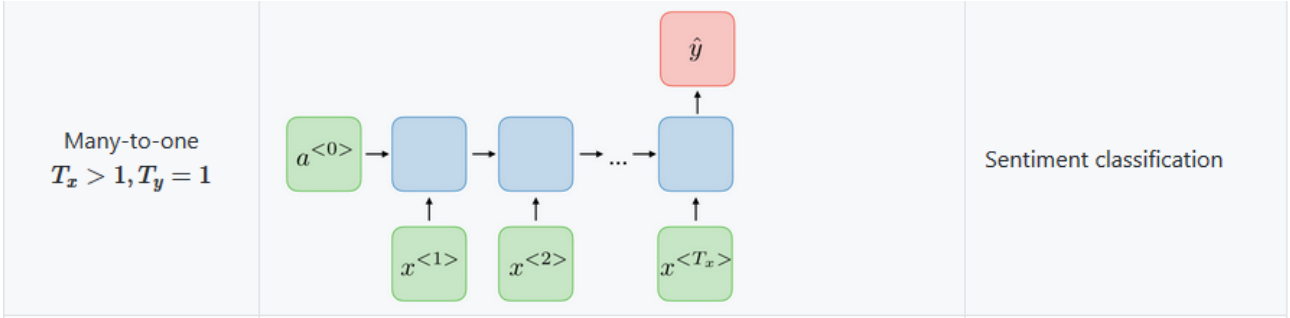
In this assignment we take a look at the Imdb Dataset and train a RNN based classifier to properly classify the the movie reviews into positive and negative. Recurrent neural networks, also known as RNNs, are a class of neural networks that allow previous outputs to be used as inputs while having hidden states. They are typically as follows:



a typical RNN has multiple types:

Type	Application
one-to-one	Traditional Neural Network
one-to-many	Music Generation
many-to-one	Sentiment Classification
many-to-many ($n_{input} = n_{output}$)	Name Entity Recognition
many-to-many ($n_{input} \neq n_{output}$)	Machine Translation

For our case, we would be classifying reviews, which is sentiment classification. This involves taking in a string of characters and classifying them.



Experimental Results

The experiment was run for 3 epochs, with 25K training examples and 25k cross validation examples of padded word length. The Adam optimizer was used to optimize the weights. The experiment was run on a simple NN network and this result was juxtaposed against an RNN made from LSTM, GRU and Bidirectional LSTM with embedding size of 128. This yielded the following results.

In the simple NN with a vocabulary size of 5000 words, max length of 2697, we got:
Training Accuracy = 96.41%
Validation Accuracy = 87.98%

In the LSTM with a vocabulary size of 5000 words, max length of 2697, we got:
Training Accuracy = 91.62%
Validation Accuracy = 86.74%

In the GRU with a vocabulary size of 5000 words, max length of 2697, we got:
Training Accuracy = 90.34%
Validation Accuracy = 86.20%

In the Bidirectional LSTM with a vocabulary size of 5000 words, max length of 2697, we got:
Training Accuracy = 90.73%
Validation Accuracy = 87.57%

following this preliminary results, we tried changing the max length to 1000:
Changing the max length of the words to 1000 for the LSTM resulted in
Training Accuracy = 91.09%
Validation Accuracy = 87.62%
Changing the max length of the words to 500 for the LSTM resulted in
Training Accuracy = 91.50%
Validation Accuracy = 87.56%

Changing the max length of the words to 1000 for the GRU resulted in
Training Accuracy = 89.88%
Validation Accuracy = 86.87%
Changing the max length of the words to 500 for the GRU resulted in
Training Accuracy = 90.52%
Validation Accuracy = 87.48%

Changing the max length of the words to 1000 for the Bidirectional LSTM resulted in
Training Accuracy = 90.92%
Validation Accuracy = 87.34%
Changing the max length of the words to 500 for the Bidirectional LSTM resulted in
Training Accuracy = 90.38%
Validation Accuracy = 88.06%

After changing the vocabulary size of the LSTM to 2500 words we got:
Training Accuracy = 88.68%
Validation Accuracy = 87.50%
After changing the vocabulary size of the LSTM to 1000 words we got:
Training Accuracy = 85.19%
Validation Accuracy = 84.93%

Changing the embedding size of LSTMs to 64, we get:
Training Accuracy = 91.12%

Validation Accuracy = 88.26%
Changing the embedding size of LSTMs to 32, we get:
Training Accuracy = 90.31%
Validation Accuracy = 86.90%

Finally we tried cascading models to see if it provided in performance benefits:

The models that were tried were cascading two LSTM blocks, cascading two GRU blocks and cascading an LSTM and a bidirectional LSTM block. the following results were found:
2 LSTM Blocks:
Training Accuracy = 90.46%
Validation Accuracy = 87.31%

2 GRU Blocks:
Training Accuracy = 87.82%
Validation Accuracy = 84.92%

1 LSTM and 1 Bidirectional LSTM:
Training Accuracy = 91.34%
Validation Accuracy = 87.70%

Inferences and Discussions

As we can see, RNN is a very powerful tool for not just sentiment classification but many different applications. From the experimental results that we found, we can say that the model is being trained properly with high enough training and testing accuracies. Changing the parameters has yielded the following observations:

1. **Reducing the max word string length** only reduces the accuracy by a little but creates a huge difference in the model training times. This is really useful as it means that we don't need the maximum word length to produce desirable results.

2. **Reducing the vocabulary size** reduces shows significant reduction in training and testing accuracies. This is possible because reducing the vocabulary size basically means that we are reducing the features for a model to train upon. Low vocabulary size may lead to predictions being biased towards certain words and may result in polarizing reviews even if the gist of the entire word string may refer to something else.

3. **Reducing the embedding size** does neither shows much reduction in the accuracy, nor does the training time decrease significantly. Further testing on with more drastic changes may be required.

4. **Cascading models:** This is an interesting subject. While there were no significant changes to our results, literature review suggests that this provides better results in EEG signal classification, Hyperspectral Image Classification (basically a one-to-many RNN model).