Assignment

Jacob Joy

Kent State University

Dr. CJ Wu

4/20/2025

Setup

The IMDB data was downloaded, unzipped, and the unsupervised data removed. The data was then split into training, validation, and testing data. 50% of each set is positive and the other 50% negative reviews. The number of validation samples is 10,000, the testing data is 25,000, and the initial training data is 100. Only the top 10,000 tokens are used, and each review is cut off at 150.

Procedure

1. After the initial setup, a bag-of-words model was created using unigrams and the results were recorded.
2. Bigrams with binary encoding are used next and the results recorded.
3. Bigrams with TF-IDF encoding is used next and the results recorded.
4. One hot encoded vector sequence is used next and the results are recorded.
5. Next, the previous model was used again but is run with an embedding layer trained from scratch.
6. The previous model was ran again but this time the addition of masking was added.
7. Next a pretrained word embedding is used to run the model
8. Next a Transformer encoder is used and the results are recorded.
9. Next, the Transformer encoder is used but with positional embedding added.
10. Steps 8 and 9 are repeated using a pretrained embedding layer
11. Repeat the steps above using different sizes for the training set

Summary

Following the information given the first set of models were built using a training set of 100 samples. While example code was used it was found that the examples were built on Keras 2 which caused some issues however, there is work arounds to convert some of the code to use Keras 3 by swapping tf.* and tf.math* with keras.ops. Right away the embedding layer performed better than the pretrained layer based on validation accuracy. 0.5764 for the embedding layer and 0.534 for the pretrained layer. This was interesting as 100 training samples seemed too small, but this could be because of the small sequence size we allowed. A training sample of 300 was used next to hit our maximum token size of 10,000. The best performing model was the bag-of-words models that produced a validation of 0.7. Throughout all training sizes used, bag-of-words performed the best.

Unfortunately, we do not have enough training data (if holding the sequence length at 150) to see if this would hold true using the rule of thumb that number of samples / mean sample length > 1500 would suggest better performance from a sequence model. Almost all models tested did beat the baseline of 50% except the pretrained transformer encoder which was only 0.3211 validation accuracy. While higher training set sizes were used to show increases in performance throughout a set of models was created using 50 samples. The 50-training sample did show that

the pretrained embedding layer does perform slightly better based on a validation of 0.5379 vs 0.5346 for the embedding from scratch.

Conclusion

In conclusion having at least 100 training samples allowed for the embedding layer trained from scratch to perform better. The following based on validation accuracy was true from the results of the experiments ran: Bag-of-words > transformer > embedding from scratch > pretrained embedding > baseline. If you have enough resources and time, it is best to train an embedding layer from scratch. Depending on what problem is being worked on and the amount of data available, pre-trained embedding may be a better option. Additional optimization could have been performed to increase the performance of the experiments such as increasing the max number of tokens as well as increasing the sequence size. This would lead to increased time. These experiments again highlight that the more training data you have generally the better your model will perform.

Model	Training Size	Training Accuracy	Training Loss	Validation Accuracy	Validation Loss	Test Accuracy
Binary_1gram	100	1	0.1726	0.7099	0.5937	0.7
Binary_2gram	100	0.9651	0.2074	0.7	0.5952	0.699
TFIDF_2Gram	100	0.8824	0.3076	0.6344	0.6699	0.625
One_hot_bidir_lstm	100	0.8363	0.6544	0.5305	0.6911	0.525
embeddings_bidir_lstm	100	0.8737	0.5505	0.5537	0.6852	0.548
embeddings_bidir_lstm_with_masking	100	0.7033	0.4959	0.5764	0.6814	0.568
glove_embeddings_sequence	100	0.5752	0.662	0.534	0.6913	0.522
transformer_encoder	100	0.9291	0.1721	0.6408	0.6607	0.638
full_transformer_encoder	100	0.9383	0.1751	0.6031	0.6721	0.599
transformer_encoder_pretrained	100	0.6006	0.7206	0.5849	0.6716	0.584
full_transformer_encoder_pretrained	100	0.4977	0.8106	0.5635	0.6815	0.561
Binary_1gram	300	0.9966	0.125	0.7773	0.4837	0.772
Binary_2gram	300	0.9966	0.0838	0.7868	0.4658	0.784
TFIDF_2Gram	300	0.8804	0.5835	0.7047	0.5767	0.699
One_hot_bidir_lstm	300	0.8063	0.5484	0.6594	0.6283	0.65
embeddings_bidir_lstm	300	0.8066	0.407	0.673	0.6402	0.671
embeddings_bidir_lstm_with_masking	300	0.8625	0.4342	0.6467	0.6287	0.568
glove_embeddings_sequence	300	0.6531	0.6413	0.5809	0.6726	0.58
glove_embeddings_sequence 300D	300	0.8141	0.465	0.6509	0.6285	0.646
transformer_encoder	300	0.8931	0.2571	0.7153	0.5689	0.719
full_transformer_encoder	300	0.872	0.2517	0.6686	0.6095	0.666
transformer_encoder_pretrained 300d	300	0.8143	0.3782	0.716	0.617	0.754
full_transformer_encoder_pretrained 300d	300	0.7892	0.5067	0.7264	0.5452	0.721
Binary_1gram	15000	0.8848	0.3028	0.8873	0.2875	0.885
Binary_2gram	15000	0.898	0.2743	0.8957	0.2801	0.891
TFIDF_2Gram	15000	0.8467	0.3694	0.8892	0.2964	0.882
One_hot_bidir_lstm	15000	0.8719	0.3355	0.8047	0.4071	0.781
embeddings_bidir_lstm	15000	0.8591	0.349	0.8048	0.4277	0.785
embeddings_bidir_lstm_with_masking	15000	0.8695	0.3167	0.8466	0.3719	0.835
glove_embeddings_sequence	15000	0.8557	0.3411	0.8274	0.3974	0.831
transformer_encoder	15000	0.8657	0.3165	0.8345	0.372	0.83
full_transformer_encoder	15000	0.7977	0.438	0.8403	0.3749	0.83
transformer_encoder_pretrained	15000	0.8199	0.4025	0.3211	0.3901	0.813
Binary_1gram	50	0.9762	0.2649	0.6196	0.652	0.622
Binary_2gram	50	0.9525	0.2514	0.613	0.6538	0.618
TFIDF_2Gram	50	0.7508	0.528	0.5057	1.855	0.505
One_hot_bidir_lstm	50	0.9258	0.654	0.5293	0.6917	0.526
embeddings_bidir_lstm	50	0.9125	0.576	0.5364	0.6889	0.532
embeddings_bidir_lstm_with_masking	50	1	0.5061	0.5346	0.6943	0.525
glove_embeddings_sequence	50	0.5742	0.676	0.5379	0.6934	0.535

Table of results. Red highlights show when pretrained performed better; Green highlights show from scratch performing better; Yellow highlight shows best performing model overall.