Assignment 3

# Training Models Across Different Sample Sizes:

The values are set to:

Cutoff reviews set to 150 words. training samples = 100

Validate samples = 10,000 words= 10,000

The models underwent training using varying sample sizes ranging from 100 to 10,000. The test loss and accuracy of each model are documented in the table below.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| sample size | one hot encoded sequence | | Embedded | | Embedded masked | | pre trained | |
|  | Test Loss | Test Accuracy | Test Loss | Test Accuracy | Test Loss | Test Accuracy | Test Loss | Test Accuracy |
| 100 | 0.6218 | 0.6629 | 0.6708 | 0.5857 | 0.6586 | 0.608 | 0.6787 | 0.6134 |
| 500 | 0.697 | 0.565 | 0.7138 | 0.6067 | 0.7351 | 0.6192 | 0.6193 | 0.6696 |
| 2000 | 0.6596 | 0.5992 | 0.7226 | 0.7108 | 0.8353 | 0.7032 | 0.5391 | 0.7248 |
| 5000 | 0.4891 | 0.7961 | 0.5375 | 0.7924 | 0.765 | 0.7645 | 0.5137 | 0.7836 |
| 10000 | 0.4380 | 0.801 | 0.4455 | 0.798 | 0.4349 | 0.811 | 0.4573 | 0.783 |

# Train sample 100, Validation 10000:

## Initial Setup:

## The IMDB review dataset utilized in this assignment has been imported.

1. In the beginning, the model was configured by collecting 100 training samples, with each review consisting of a maximum of 150 words in length. In total, 10,000 words were utilized as input for the model.
2. Additionally, 10,000 validation samples of both positive and negative reviews are used to verify this model.
3. The loss function "binary cross-entropy" was employed in this classification model, which utilized the "Adam" optimizer.

## Models Trained:

1. Four models were trained, validated, and evaluated utilizing the initial configuration and accuracy performance metrics.
2. A single model employing one hot-encoded sequences has demonstrated a test accuracy of 0.801 and a test loss of 0.4380.
3. The test accuracy for the embedded model without masking was 0.4455 and the test loss was 0.798.
4. 1. An e mbedd model that implemented masking achieved an accuracy of 0.4349 and a test loss of 0.811.
5. 1. Pre-trained model. Global Vectors for Word Representation (GloVe) provided a test

loss of 0.4573 and test accuracy of 0.783.

The findings revealed that RNNs with embedded layers outperformed alternative word embedding approaches, such as one-hot encoded sequences, in sentiment analysis. The embedded layer-based models regularly beat other methods in terms of test loss and accuracy.

Furthermore, several types of embedded layers, such as standard and masked embedded layers, are compared. Standard embedded layer-based models performed somewhat better in terms of test accuracy than masked embedded layers. Although the masking technique allows the model to ignore padding tokens and focus only on the actual word embeddings, resulting in more meaningful representations and improved performance in this model implementation, it is clear that masking has no effect on the given IMDb dataset.

CONCLUSION:

1. For all cutoff reviews and training sample sizes, the embedding layer model's validation accuracy exceeds its test accuracy. This indicates that the model may be overfitting to the training data.   
   • The pre-trained model has greater validation accuracy than test accuracy for certain cutoff reviews and training sample sizes, but lower accuracy for others. This shows that the model's performance varies more than the embedding layer model.
2. The study found that the plain embedding layer model outperformed the pre-trained model, contradicting the popular belief that pre-trained embeddings improve model performance. In general, it is crucial to note that the pre-trained model in this case is not optimized for the job at hand, and the embeddings were not fine-tuned during training. Essentially, fine-tuning the embeddings may lead to improved performance.
3. Conclusions should be drawn with caution due to the limited number of training samples and hyperparameters used. It is likely that changing hyperparameters or more training data will result in different conclusions.