ASSIGNMENT 3

IMDB MOVIE REVIEW ANALYSIS USING AN EMBEDDING LAYER AND A PRE-TRAINED EMBEDDING LAYER

Executive Summary:

This task involves a binary classification problem using the IMDB dataset, where the objective is to determine whether a given movie review expresses a positive or negative sentiment. The dataset contains 50,000 reviews, and we limit the reviews to 150 words for processing. We also restrict the number of training samples to 100, 500, 1000, or 10000, and validate on 10000 samples. Additionally, we only consider the top 10000 words in the dataset. Pre-processing techniques are applied to the data before feeding it into both an embedding layer and a pre-trained embedding model. We assess the performance of different approaches to determine the best performing model.

Problem Statement:

The IMDB dataset presents a binary classification challenge, where the task is to classify movie reviews as either positive or negative. To evaluate the effectiveness of different approaches, we test multiple models and compare their performance. The primary goal is to identify the approach that yields the best results.

Data-Preprocessing:

The IMDB dataset comprises movie reviews that are labeled as positive or negative based on sentiment. To prepare the dataset for input into a neural model, each review is transformed into a sequence of word embeddings, where each word corresponds to a vector of a fixed size. The vocabulary size is limited to 10,000. The original sequence of words is converted into a sequence of integers, with each integer representing a different word. However, these integers are not suitable for input into a neural model, so they must be transformed into tensors. One way to achieve this is by creating a tensor with integer data type and shape (samples, word indices), where each sample is of equal length. This requires padding each review with dummy words (integers) to ensure they are the same length.

Model Building & Evaluation Process:

This study explores two distinct approaches to creating word embeddings for the IMDB review dataset: a custom-trained embedding layer and a pretrained word embedding layer using the GloVe model. The GloVe model is a widely used pre-trained word embedding model that is capable of capturing semantic and syntactic relationships between words, making it a popular choice for natural language processing tasks. In this study, the 6B version of the GloVe model was used, which includes 6 billion tokens and 400,000 words, trained on a combination of Wikipedia data and Gigaword 5.

To evaluate the effectiveness of different embedding techniques, we implemented two embedding layers on the IMDB review dataset. The first layer was a custom-trained embedding layer, trained on

different samples of the dataset and evaluated using a testing set. The second layer used a pre-trained word embedding layer from GloVe and was also tested on varying sample sizes. We compared the accuracies of both models to determine which approach yielded better results.

Findings:

Custom-Trained Embedding Layer Model Performance:

The custom-trained embedding layer demonstrated a high level of accuracy in the range of 97% to 100%, depending on the size of the training sample used. The highest accuracy was achieved when the training sample size was set to 100. One possible explanation for this high level of accuracy is that the embedding layer is specifically designed for the task of IMDB review sentiment classification, resulting in more effective text data representations.

However, it is also important to note that there was no significant improvement in accuracy beyond a training sample size of 100, indicating that the benefits of using additional training data may be limited for this technique.

Pre-Trained Word Embedding Layer (GloVe) Model Performance:

The accuracy of the pretrained word embedding layer (GloVe) varied between 70% and 94%, depending on the size of the training sample. The best performance was achieved with a training sample size of only 1000. One possible explanation for this high accuracy with a small training sample is that the pretrained embeddings contain a lot of the semantic information in the text, which makes them effective even with limited training data. In contrast, the custom-trained embedding layer achieved accuracy between 97% and 100%, with the best performance obtained with a training sample size of 100. The custom-trained embeddings were specifically trained for the task at hand, which may have resulted in more effective representations of the text data. However, the accuracy did not improve significantly beyond a training sample size of 100, indicating that additional training data may not provide substantial benefits for this technique.

Results in the form of table:

Custom-Trained Embedding Layer Model				
Sample Size	100	500	1000	10000
Accuracy (%)	100	97.5	98	98.04
Pre-Trained Word Embedding Layer (GloVe) Model				
Sample Size	100	500	1000	10000
Accuracy (%)	91.98	94.80	97	70.88

Conclusion:

In conclusion, as the training sample size grows, the effectiveness of pretrained embeddings in capturing the nuances of the task may decrease, resulting in lower accuracy. Furthermore, the prompt notes that using pretrained embeddings with larger training sample sizes leads to overfitting, which lowers the accuracy. Consequently, it's challenging to conclude which technique is superior because it depends on the requirements and limitations of the task. However, the custom-trained embedding layer typically outperformed the pretrained word embedding layer in this study, particularly with larger training sample sizes. If resources are scarce, and a small training sample size is required, the pretrained word embedding layer may be a better option, but care must be taken to prevent overfitting.