ASSIGNMENT 3

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

Executive Summary:

In this task, we aim to classify whether a movie review expresses positive or negative sentiment using the IMDB dataset. The dataset comprises 50,000 reviews, but we limit each review to 150 words for efficient processing. We use 100, 500, 1000, or 10000 training samples and validate 10000 samples. Our analysis only considers the top 10000 words in the dataset, and we apply pre-processing techniques to the data before feeding it into an embedding layer and a pre-trained embedding model. We evaluate the effectiveness of various approaches to identify the best-performing model.

Problem Statement:

The IMDB dataset poses a binary classification challenge in which movie reviews are classified as positive or negative. Our objective is to test multiple models and compare their performance to identify the most effective approach. Ultimately, our aim is to determine the optimal method for achieving the best results.

Data-Preprocessing:

The IMDB dataset includes movie reviews that are classified as positive or negative based on their sentiment. To prepare this dataset for input into a neural model, we convert each review into a sequence of word embeddings, where each word corresponds to a fixed-size vector. We limit the vocabulary size to 10,000, and then represent each word in the review as an integer. However, these integers are not directly compatible with neural models, so we transform them into tensors. One way to accomplish this is by creating a tensor with an integer data type and shape (samples, word indices) that has samples of equal length. To achieve equal length, we pad each review with dummy words (integers).

Model Building & Evaluation Process:

This study investigates two approaches for creating word embeddings on the IMDB review dataset. The first approach is a custom-trained embedding layer, while the second approach employs a pre-trained word embedding layer using the widely used GloVe model. The GloVe model is known for its ability to capture semantic and syntactic relationships between words, making it a popular choice for natural language processing tasks. The study utilized the 6B version of the GloVe model, which has 6 billion tokens and 400,000 words, trained on a combination of Wikipedia data and Gigaword 5.

To evaluate the effectiveness of these two embedding techniques, the study implemented both embedding layers on the IMDB review dataset. The custom-trained embedding layer was trained on different samples of the dataset and evaluated using a testing set, while the pre-trained GloVe word

embedding layer was also tested on varying sample sizes. The accuracies of both models were compared to determine which approach yielded better results.

Findings:

Custom-Trained Embedding Layer Model Performance:

The results of the study indicated that the custom-trained embedding layer exhibited a high degree of accuracy, ranging from 97% to 98%, depending on the size of the training sample used. The highest accuracy was achieved with a training sample size of 1000. One possible explanation for the high level of accuracy is that the custom-trained embedding layer was specifically designed for the task of IMDB review sentiment classification, resulting in more effective text data representations.

However, it is worth noting that the study found no significant improvement in accuracy beyond a training sample size of 1000, suggesting that the benefits of using additional training data may be limited for this technique.

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

According to the study, the accuracy of the pretrained word embedding layer (GloVe) varied between 92% and 100%, depending on the size of the training sample, with the best performance obtained with a training sample size of only 100. The study suggests that one possible explanation for the high accuracy with a small training sample is that the pretrained embeddings contain a significant amount of semantic information in the text, making them effective even with limited training data.

In contrast, the custom-trained embedding layer achieved accuracy between 97% and 98%, with the best performance obtained with a training sample size of 1000. The custom-trained embeddings were specifically designed for the task of IMDB review sentiment classification, which may have resulted in more effective representations of the text data.

However, the study found that the accuracy did not improve significantly beyond a training sample size of 1000 for the custom-trained embedding layer, indicating that additional training data may not provide substantial benefits for this technique. Overall, the study suggests that both approaches to creating word embeddings can be effective for sentiment classification of IMDB reviews, with their relative effectiveness depending on the size of the training sample and the specific characteristics of the dataset.

Results in the form of table:

Custom-Trained Embedding Layer Model				
Sample Size	100	500	1000	10000
Accuracy (%)	97	97	98	98
Pre-Trained Word Embedding Layer (GloVe) Model				
Sample Size	100	500	1000	10000
Accuracy (%)	100	100	95	93

Conclusion:

In conclusion, the choice between custom-trained embeddings and pretrained word embeddings depends on the specific needs of the task and the resources available. In this study, the custom-trained embeddings generally outperformed the pretrained embeddings, especially with larger training sample sizes. However, the pretrained embeddings may be a good choice when working with limited training data. Using pretrained embeddings with larger training sample sizes can lead to overfitting, which reduces accuracy. Thus, the most appropriate technique will depend on the task's requirements and the available resources.