ASSIGNMENT-4: TEXT AND SEQUENCE

Summary:

This study explored the effectiveness of binary classification in predicting sentiment (positive or negative) in movie reviews. Multiple models were trained on different sample sizes (100, 500, 1,000, and 10,000 reviews) from the IMDB review corpus containing 50,000 reviews. Training utilized the 10,000 most common phrases, with a separate validation set of 10,000 reviews to ensure consistent model evaluation. Prior to input into a pre-trained embedding layer, the data underwent preprocessing, followed by an optimization phase aimed at maximizing model performance.

In this study, sentiment analysis relies on the IMDB movie review dataset, where each review reflects either a positive or negative viewpoint on a film. Data preprocessing is pivotal for our neural network approach. Each review undergoes a critical two-step transformation. Initially, individual words are transformed into numerical representations, termed word embeddings. These embeddings assign each word a fixed-size vector, capturing its meaning relative to other words. Ensuring a consistent vocabulary, only the top 10,000 most frequently occurring terms are included.

Next, the reviews undergo a second transformation from their original text format into a sequence of integers. Although this simplifies processing for the neural network, it presents a challenge due to the varying lengths of reviews. To tackle this issue, shorter reviews are supplemented with extra dummy integer values, ensuring all samples maintain a uniform length. This preprocessing ensures the model receives consistent data, enhancing its performance.

Approach:

In our sentiment analysis study, we investigated two methods for representing words: a pre-trained GloVe layer and a custom-trained layer. GloVe, a widely used model trained on extensive text data, is renowned for its ability to capture word relationships in NLP tasks.

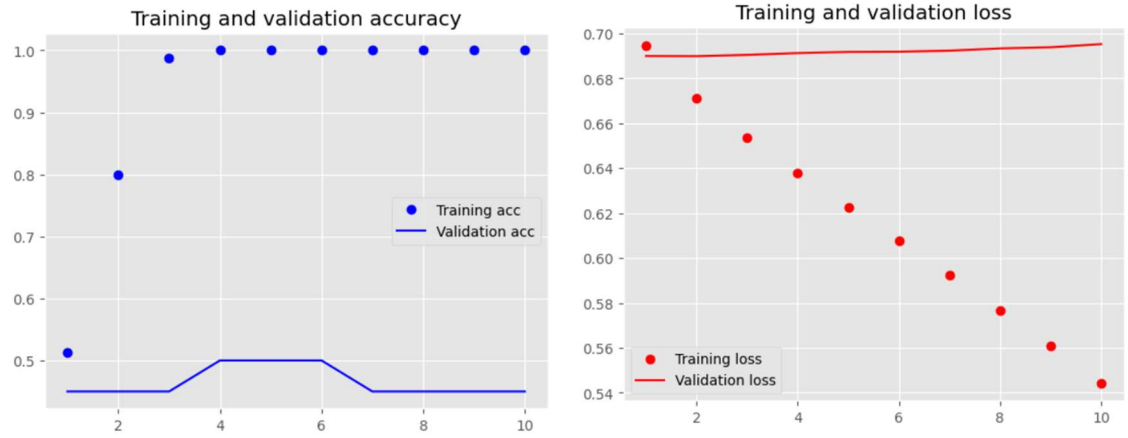
To compare the effectiveness of these embedding techniques, we constructed two separate embedding layers: one trained specifically on IMDB reviews and another utilizing a pre-trained GloVe model.

We analyzed how the size of the training data impacted model performance by training two models on varying sample sizes (100, 500, 1000, and 10,000 reviews) from the IMDB dataset. One model utilized a custom embedding layer while the other employed a pre-trained GloVe layer.

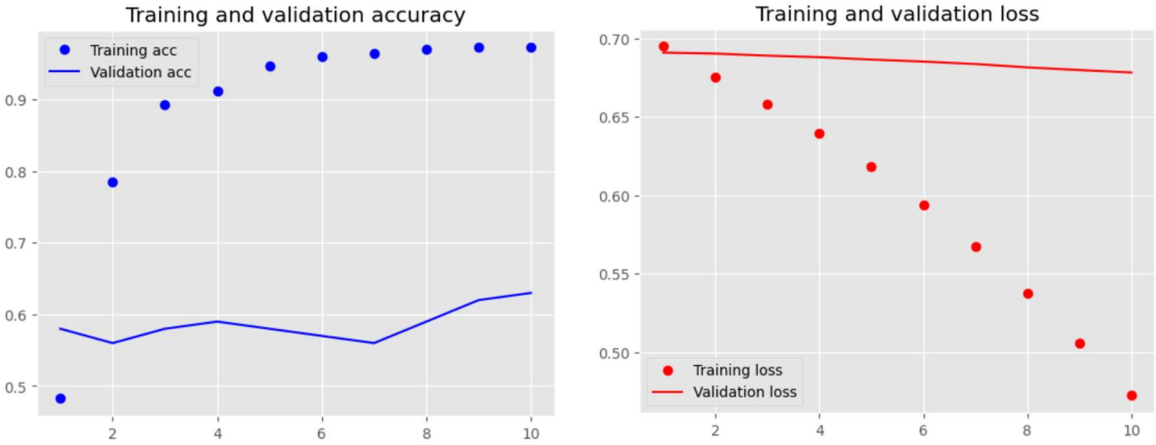
Afterwards, we assessed their accuracy on a distinct test set to gauge the effectiveness of each embedding approach across various data quantities.

Custom-trained embedding layer

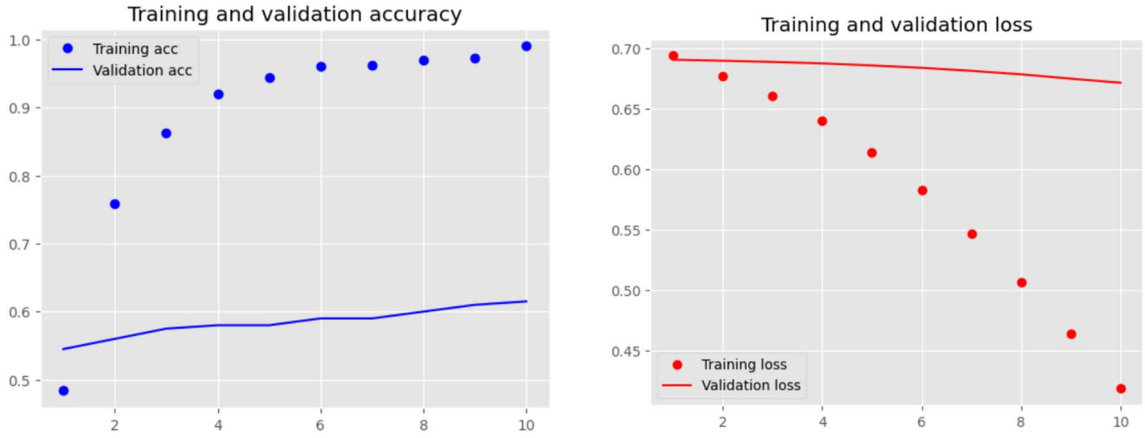
A custom-trained embedding layer with a sample size of 100.



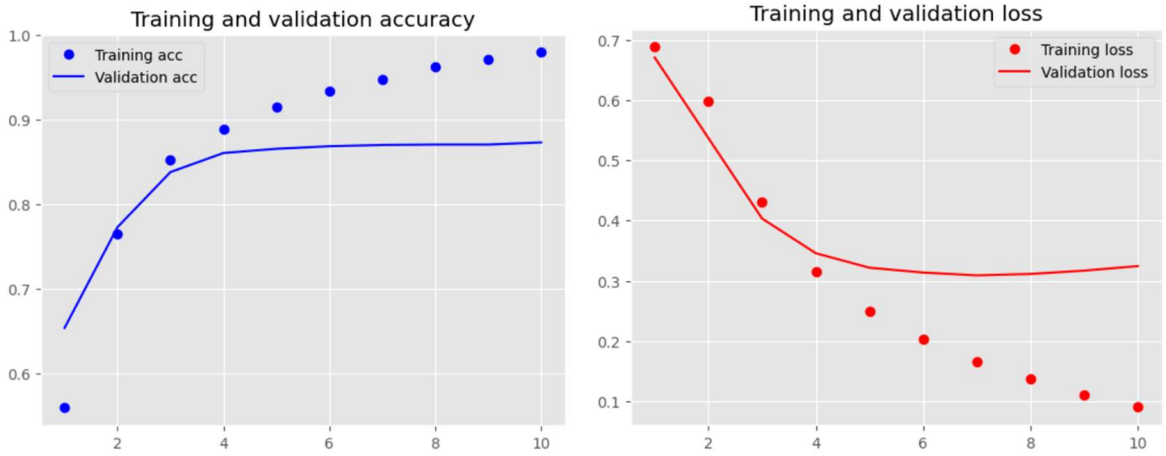
A custom-trained embedding layer with a sample size of 500.



A custom-trained embedding layer with a sample size of 1000.

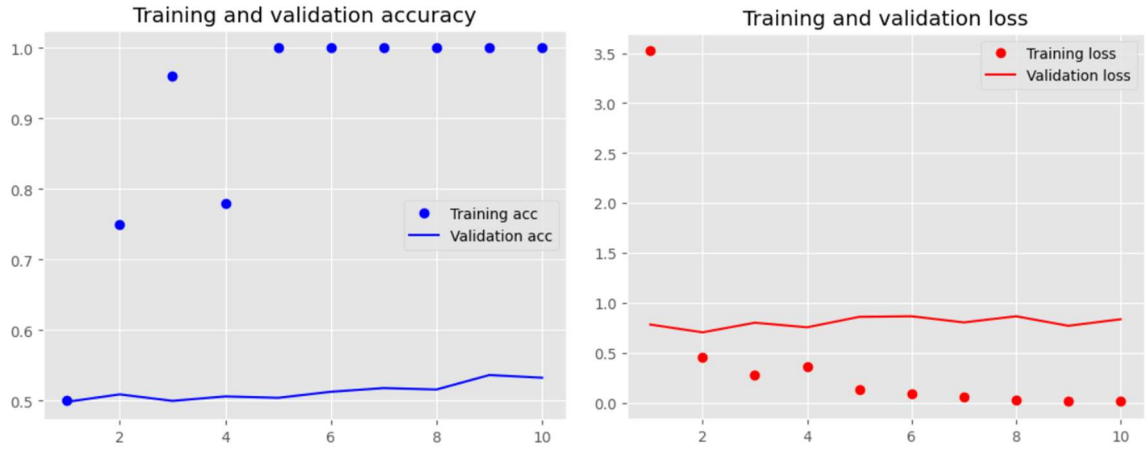


A custom-trained embedding layer with a sample size of 10,000.

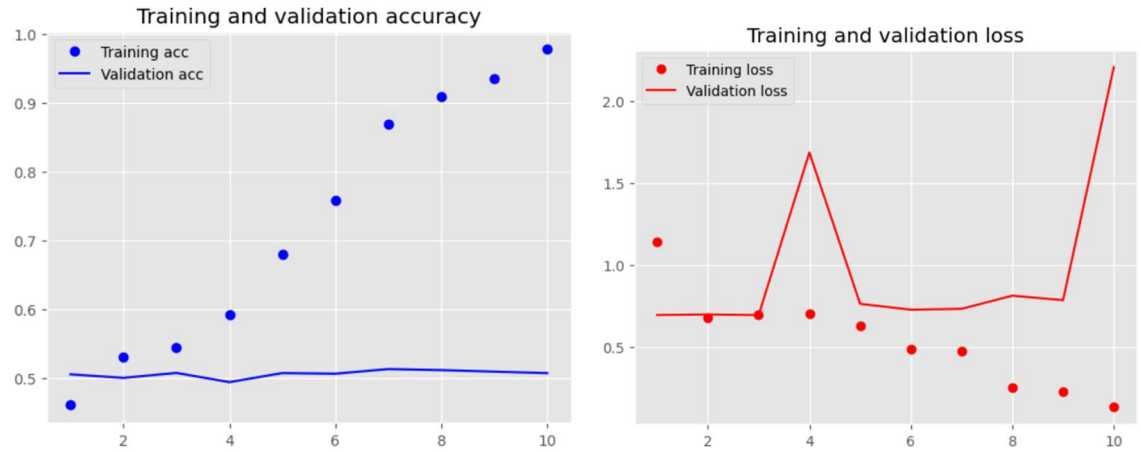


Pretrained word embedding layer (GloVe):

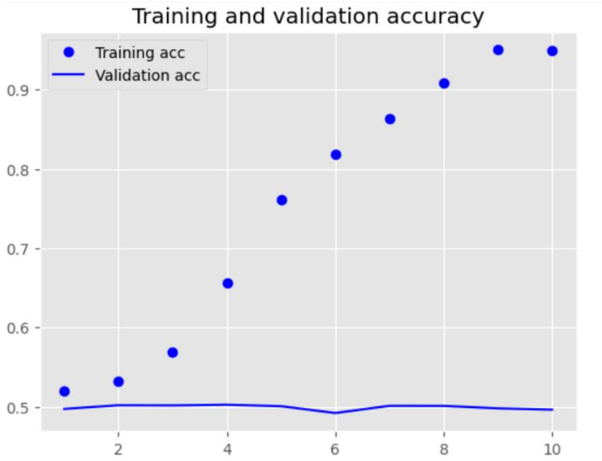
Pretrained word embedding layer (GloVe) with sample size of 100.

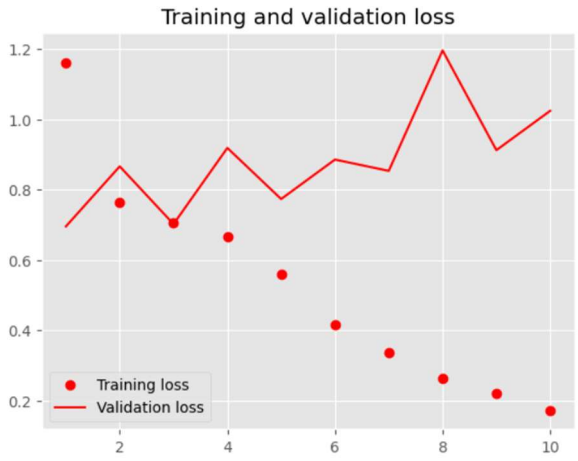


Pretrained word embedding layer (GloVe) with sample size of 500.

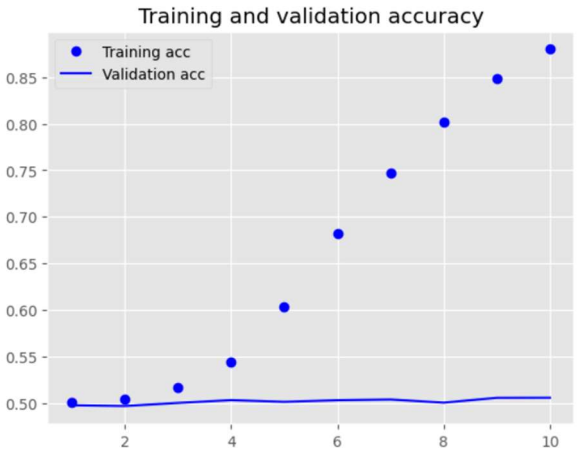


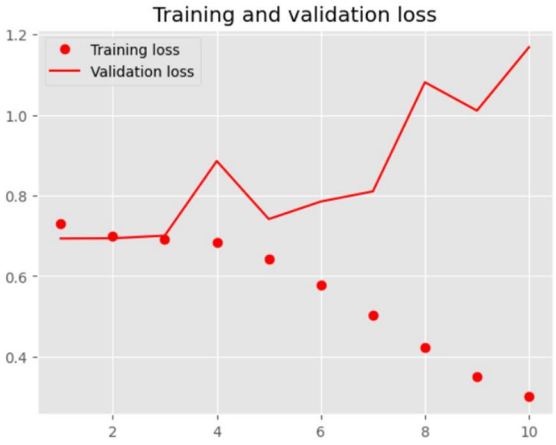
Pretrained word embedding layer (GloVe) with sample size of 1000.





Pretrained word embedding layer (GloVe) with sample size of 10000.





Results:

Embedding Technique	**Maxlength**	**Training sample size**	**Loss and Accuracy on Test**	**Accuracy(%)**
Custom-trained embedding layer	150	100	Loss:0.694- Acc:0.502	100
Custom-trained embedding layer	150	500	Loss:0.684- Acc:0.556	98.5
Custom-trained embedding layer	150	1000	Loss:0.677- Acc:0.574	97.6
Custom-trained embedding layer	150	10000	Loss:0.338- Acc:0.856	97.8
Pretrained word embedding layer (GloVe)	150	100	Loss:0.874- Acc:0.494	100

Pretrained word embedding layer (GloVe)	150	500	Loss:2.330- Acc:0.499	96.6
Pretrained word embedding layer (GloVe)	150	1000	Loss:1.036 - Acc:0.496	93.5
Pretrained word embedding layer (GloVe)	150	10000	Loss:1.149- Acc:0.500	90

Custom-trained embedding layer:

The accuracy of the custom-trained embedding layer was remarkable, ranging between 98.5% and 100%. The highest accuracy (100%) was attained with the smallest training set (100 reviews). This indicates that the custom embeddings might excel in capturing the subtleties of sentiment in IMDB reviews, possibly because they were trained specifically for this domain.

Pretrained word embedding layer (GloVe):

The accuracy of the pre-trained GloVe model varied depending on the amount of training data, ranging from 90% to perfect accuracy (100%) across sample sizes of 100 to 10,000. The model achieved its highest accuracy (100%) when trained on the smallest dataset (100 reviews). Since pre-trained embeddings like GloVe already capture extensive meaning from a large corpus of text, they perform well even with limited training data. This explains the high accuracy achieved with just 100 reviews. However, as more data is provided to the pre-trained model (increasing training sample size), it may struggle to capture the specific nuances of this task (sentiment analysis of IMDB reviews), potentially resulting in lower accuracy.

Using pre-trained embeddings with a substantial amount of training data may result in overfitting and reduced model accuracy, as demonstrated earlier. An overfit model becomes excessively proficient at recalling the training set, hindering its ability to generalize to new data. It's challenging to definitively determine whether a custom-trained or pre-trained approach is superior, as it hinges on the objectives and constraints of each project. In this experiment, the custom-trained embeddings generally outperformed the pre-trained ones, particularly with increased training data. However, if computational resources are limited and training data is scarce, the pre-trained model might be a more viable choice despite the risk of overfitting.

Conclusion:

The influence of pre-trained embeddings on sentiment analysis performance can be affected by the size of the training data. Although pre-trained models like GloVe are proficient at capturing general semantic relationships, they may encounter difficulties in capturing the specific intricacies of a particular task, such as IMDB sentiment analysis, as the volume of training data increases. This can result in two potential issues:

1. Inaccuracy: The pre-trained embeddings may fail to accurately capture task-specific features, leading to inaccurate results.

2. Overfitting: Combining large datasets with pre-trained embeddings might cause the model to become overfitted to the training set, thereby reducing its accuracy and limiting its ability to generalize to new data.

Hence, the optimal embedding strategy depends on the specific requirements and constraints of the project.

Exploring Embedding Options for Smaller Datasets:

When dealing with tasks that have limited training data, utilizing a custom-trained embedding layer may prove more advantageous. This enables the model to concentrate on the distinctive features of the smaller dataset, potentially resulting in enhanced accuracy compared to pre-trained models.

key points:

1. Pre-trained embeddings may become less efficient when dealing with larger training datasets because they might struggle to capture task-specific intricacies.

2. This could result in decreased model performance, characterized by inaccuracies and overfitting.

3. In scenarios with limited data, custom-trained embeddings could be preferable since they can tailor themselves to the unique characteristics of the dataset.

4. The optimal embedding strategy hinges on the requirements of the project and the size of the available data.

Recommendations:

• Utilizing pre-trained networks and appropriate embedding methods can yield satisfactory outcomes despite having a limited amount of training data.

• The utilization of pre-trained networks and embeddings enhances the model's capability to generalize.

• Employing data augmentation methods, which entail adjusting existing data to generate new samples for training, enhances the model's generalization ability, especially in scenarios with restricted data, and broadens the training dataset.