Assignment - 4
Advance Machine learning

SENTIMENT ANALYSIS ON IMDB REVIEWS USING WORD EMBEDDING

Introduction:

This research is focused on investigating word embedding approaches for sentiment analysis on the IMDB dataset. The IMDB dataset contains 50,000 movie evaluations, half of which are positive and half of which are negative. The dataset is divided into 25,000 training reviews and 25,000 testing reviews. We stopped reviewing at 150 words and limited training samples to 100. We verified 10,000 samples and only took into account the top 10,000 terms. The overall report aims to compare the effectiveness of different models with different training samples and embedding layers. The models in this paper were trained using a bidirectional LSTM architecture.

Methodology:

Basic Sequence Model: To begin, we trained a basic sequence model to create a baseline performance for the assignment. The model obtained high training accuracy, but validation accuracy was somewhat lower, indicating overfitting to the validation set.

Embedding Layer from Scratch: Next, we trained a model from scratch that employs word embedding without activating masking. The model achieved higher training accuracy but lower validation accuracy, indicating overfitting to the training set. We also discovered that activating masking can reduce overfitting and increase the model's capacity to handle variable-length sequences.

Embedding Layer from Scratch with Masking: We trained a model with masking enabled that outperformed the prior model in terms of validation accuracy. This implies that masking is an important element to consider when utilizing word embedding.

Pretrained Word Embedding: We taught a model utilizing pre-trained GloVe word embedding, though the training accuracy was worse than that of all prior models, implying that the pre-trained model did not capture the subtleties and context of the dataset. This emphasizes the significance of experimenting with alternative embeddings or fine-tuning the pre-trained model for improved performance.

Different Training Samples: Finally, we experimented with different training sample sizes to discover the best size for training the embedding layer. We discovered that the model worked best with 1000 training samples, delivering high training and validation accuracy while preserving low training and validation loss.

Results:

Model	Train Accuracy%	Valid Accuracy%	Train_loss	Valid_loss	Test Accuracy %
Basic Sequence	0.9559	0.8003	0.1440	0.4794	0.809
Embedding layer from Scratch	0.9856	0.7923	0.0531	0.7717	0.800

Embedding layer from Scratch with Musking	0.9884	0.7920	0.0317	0.6128	0.811
Pretrained word Eembedding	0.8138	0.7753	0.4128	0.5401	0.768
1000 Training samples	0.9885	0.8230	0.0357	0.7131	0.806
5000 Training samples	0.9887	0.7150	0.0339	1.4748	0.796
10000 Training samples	0.9923	0.8120	0.0264	0.7839	0.794
15000 Training samples	0.9910	0.8270	0.0320	0.7448	0.803
20000 Training samples	0.9910	0.8090	0.0281	0.6564	0.801
25000 Training samples	0.9905	0.8010	0.0309	0.7048	0.801

According to the data, the model with the highest test accuracy is "Basic Sequence" with a test accuracy of 0.809. However, despite employing substantially fewer training examples, the "1000 Training samples" model also performed well, with a test accuracy of 0.806. When determining which model is the best match for a given job, it is critical to examine the trade-offs between model performance and training resources. If computing resources and time are limited in this instance, the "1000 Training samples" model may be a reasonable alternative. If more precision is necessary and more resources are available, the "Basic Sequence" model may be a better option. Finally, we discovered that utilizing a higher training sample size (10,000) produced the best results for all models.

Conclusion:

From the findings, we may infer that building the embedding of words from scratch can increase the model's training accuracy. To avoid overfitting, suitable methods for regularization such as masking should be used. Pre-trained models might not always perform well on certain datasets, suggesting the importance of experimenting with various embeddings or fine-tuning the embedding for improved performance. The size of training samples influences the performance of the layer that embeds. Word embedding is an essential approach in natural language processing, and its effectiveness is affected by a number of factors, including the size of the training dataset, regularization techniques, and pre-trained models. The outcomes of this study can be utilized to improve model performance in sentiment analysis and other NLP applications. We discovered that 1000 is the best training sample size for training the embedding layer. This project gave a thorough grasp of word embedding techniques and their use in sentiment analysis.