**Text and Sequence Data**

**Name: Priyanka Chinthakuntla**

**Student Id: 811317866**

**Introduction:**

The following project aims to apply text classification models using RNNs on the IMDB movie review dataset. For this purpose, two different word embedding techniques have been used to evaluate their performance:

1. Custom Trainable Embedding Layer
2. Pre-trained GloVe Embeddings

The assignment examines how those methods perform about specific constraints, such as a limited number of training data, truncated reviews, and a fixed vocabulary size. Evaluation metrics such as Mean Squared Error (MSE) and Root Mean Squared Error (RMSE) are used to analyze model predictions.

**Objectives:**

* To preprocess and prepare text data for RNN-based classification.
* To implement and compare custom and pretrained word embeddings.
* To evaluate the models using validation accuracy, MSE, and RMSE.
* To determine the optimal training sample size for effective model performance.

**Overview of Dataset**

The IMDB Movie Reviews Dataset is a benchmark dataset for sentiment classification. There are 50,000 movie reviews, divided into two classes: positive (1) and negative (0); 25,000 training samples and 25,000 test samples.

* **Preprocessing:** Reviews are tokenized into integer sequences, where each integer represents a word in a predefined index.
* **Vocabulary Limit:** Only the top 10,000 most frequent words are considered for this assignment.
* **Truncation and Padding:** Reviews are padded or truncated to a maximum length of 150 words.

This dataset is one of the most used for binary sentiment analysis and represents an excellent test bench for your custom-pre-trained word embeddings.

**Assignment Parameters:**

**Word Limit:** Reviews are truncated or padded to a maximum of 150 words to make the input length consistent.

Training Samples: The training set is limited to only 100 samples to artificially create limited data availability.

**Validation Samples:** The model is validated on 10,000 samples in total.

Vocabulary Size: Only the 10,000 most frequent words are kept in the vocabulary, replacing less frequent words with a placeholder token.

**Embedding Approaches:**

**Custom Embedding Layer:** An embedding layer that is trained and learns the word representations during training.

**Pre-trained GloVe Embeddings:** Non-trainable embeddings loaded from the glove.6B.50d.txt file; utilizing semantic knowledge from external corpora.

**Model Architecture:**

Two separate models are implemented to evaluate text classification performance:

1. **Custom Embedding Model**:

**Embedding Layer**: Trainable embedding layer that learns word representations directly from the dataset during training.

**Recurrent Layer**: A Bidirectional LSTM layer with 32 units to capture sequential patterns and context in the text.

**Output Layer**: A Dense layer with a sigmoid activation function for binary classification (positive or negative sentiment).

1. **Pretrained Embedding Model**:

**Embedding Layer**: Pretrained GloVe embeddings glove.6B.50b.txt are loaded and used as fixed weights in the embedding layer.

**Recurrent Layer**: A Bidirectional LSTM layer with 32 units, identical to the custom embedding model for a fair comparison.

**Output Layer**: A Dense layer with a sigmoid activation for binary classification.

**Results:**

**Model Comparison:**

Both models were trained for **10 epochs** and evaluated on, MSE, and RMSE.

|  |  |  |
| --- | --- | --- |
| **Model** | **MSE** | **RMSE** |
| Custom Embedding Model | 0.4284 | 0.4565 |
| Pretrained GloVe Model | 0.4937 | 0.7026 |

**Visualization:**

Plots comparing the performance of custom and pretrained embeddings across epochs.

**A graph with blue and orange lines

Description automatically generated**

**A graph with a line and a blue line

Description automatically generated**

A bar chart showing validation accuracy for different training sample sizes.

**A graph with a number of lines

Description automatically generated with medium confidence**

**Conclusion:**

In this assignment, we have compared custom embeddings with pre-trained GloVe embeddings for performing sentiment classification on the IMDB dataset. Pretrained embeddings outperformed custom embeddings, especially with limited training data, due to the rich semantic information they provide. However, their performance compares well with that of custom embeddings as the increase in the size of the training sample shows their great potential when enough data is available.

For both the models, accuracy, MSE, and RMSE were calculated, and from these values, it was found that pre-trained embeddings perform better. This task therefore gave sufficient proof that a pre-trained embedding is best for a small dataset, whereas a custom embedding does well when the dataset increases in size.

This study sets a base with instructive insights into how the embedding choices may affect model performance and allows for further experimentation with different architectures and larger datasets.