Assignment 4

This document presents the implementation and evaluation of multiple text classification models for sentiment analysis using Convolutional Neural Networks (CNNs) and Embedding layers. The models are trained on the Sentiment140 dataset to classify the sentiment of tweets into positive or negative categories.

1. Introduction

The code implements and compares various CNN architectures for text classification tasks using the Sentiment140 dataset. Each model is trained to classify the sentiment of tweets based on their textual content.

2. Dataset Preprocessing

The Sentiment140 dataset is loaded and preprocessed to remove unnecessary columns and clean the text data.

Text preprocessing involves removing URLs, mentions, special characters, and stopwords, followed by tokenization and lemmatization.

3. Model Architectures

Multiple CNN Architectures:

Five CNN architectures are implemented, varying in the number of convolutional layers, filter sizes, and pooling layers.

Each model consists of an Embedding layer, followed by Conv1D layers with ReLU activation, max-pooling layers, Dense layers with ReLU activation, dropout layers, and a final Dense layer with sigmoid activation for binary classification.

4. Model Training and Evaluation

The models are compiled with binary cross-entropy loss and the Adam optimizer.

Training is performed on the preprocessed training data with a batch size of 64

and 5 epochs.

Evaluation metrics such as accuracy, precision, recall, and F1-score are

computed on the test dataset to assess the performance of each model.

5. Results

Each model's test accuracy, precision, recall, and F1-score are reported to

compare their performance.

Model performance is evaluated based on classification accuracy and the ability

to generalize to unseen data.

6. Conclusion

The implemented CNN architectures demonstrate varying performance in

sentiment analysis tasks on the Sentiment140 dataset.

Evaluation metrics provide insights into each model's strengths and

weaknesses, guiding the selection of the most suitable architecture for

sentiment classification tasks, Model 2 seems to be the best performing among the provided models, with relatively high precision, recall, and F1 score. Further fine-tuning and experimentation could potentially improve the performance of all models, especially addressing issues such as overfitting in deeper architectures

7. Results

Precision for Model 1: 0.7774413061743503

Recall for Model 1: 0.776828125

F1 Score for Model 1: 0.7766808475772624

Precision for Model 2: 0.7818304855343212

Recall for Model 2: 0.781596875

F1 Score for Model 2: 0.7815648270196535

Precision for Model 3: 0.735982690856904

Recall for Model 3: 0.73529375

F1 Score for Model 3: 0.735063926914758

Precision for Model 4: 0.6648366747063427

Recall for Model 4: 0.655921875

F1 Score for Model 4: 0.650941674897976

Precision for Model 5: 0.2484212503515625

Recall for Model 5: 0.49841875

F1 Score for Model 5: 0.33157787214230006