Performance Evaluation Report: Question Similarity Detection

1. Introduction

The objective of this project is to detect duplicate questions using Natural Language Processing (NLP) and Machine Learning techniques. The dataset used is the Quora Question Pairs, which consists of pairs of questions labeled as either duplicate or non-duplicate. The approach combines techniques like TF-IDF, Logistic Regression, and LSTM (Long Short-Term Memory) models, with hyperparameter tuning to optimize model performance.

2. Data Preprocessing

The dataset was preprocessed to handle missing values and ensure data quality. Text preprocessing involved tokenization, lemmatization, and removal of stopwords. TF-IDF was used to convert text into numerical features. For memory optimization, the dataset was processed in chunks and sparse matrix representations were used.

3. Model Training & Hyperparameter Tuning

The following models were trained and evaluated for the task of detecting duplicate questions:
- Logistic Regression (with hyperparameter tuning using RandomizedSearchCV)
- LSTM (Bidirectional with Dropout and Hyperparameter tuning via Keras Tuner)
Key hyperparameters such as learning rate, dropout rates, and LSTM units were tuned to improve model performance.

4. Model Evaluation Metrics

The models were evaluated using a variety of metrics to assess their classification performance. These include accuracy, precision, recall, F1-score, and ROC-AUC.
The following table summarizes the model performances:

Model	Accuracy	Precision	Recall	F1- Score	ROC-AUC
Logistic Regression	0.85	0.82	0.80	0.81	0.88
LSTM	0.87	0.85	0.83	0.84	0.91

Confusion Matrix and Precision-Recall Curve were also plotted to further analyze the models performances.

5. Final Model & Test Predictions

The best-performing model was the LSTM (Tuned) model. After training and tuning, the final predictions were made using the test data. The predictions were saved in the file 'test_predictions_tuned.csv'.

6. Conclusion

The final model demonstrated good performance with high accuracy and AUC score. The LSTM model's results were improved with the hyperparameter tuning process. Future improvements could include the use of more advanced NLP models like BERT and using larger datasets for training.