Performance Evaluation Report for Sentence Pair Classification

1. Introduction

The objective of this project is to classify pairs of sentences into one of three categories:

- **Contradiction (0):** Sentences with opposite meanings.
- **Neutral (1):** Sentences that are related but do not imply one another.
- **Entailment (2):** One sentence logically follows from the other.

This report summarizes the performance of various models implemented to solve the problem, including a baseline Random Forest, a custom Artificial Neural Network (ANN), an LSTM-based model, and a transformer-based model (BERT).

1.1.Example of NLI

Premise: *"A man is playing a guitar on stage."*

Hypotheses and Their Labels:

✅ **Entailment**: *"A musician is performing live."*
❌ **Contradiction**: *"No one is playing music."*
🔸 **Neutral**: *"A crowd is cheering."*

2. Experimental Setup

2.1. Dataset and Preprocessing

- **Dataset:**
  The provided dataset (`train.csv`) was split into training and testing sets (80/20 split) to simulate an external test set. The data includes:
  - `id`: Unique identifier.
  - `premise`: First sentence.
  - `hypothesis`: Second sentence.
  - `label`: Class label (0, 1, or 2).
- **Text Preprocessing:**

    - **Tokenization & Normalization:** Text is converted to lowercase, punctuation is removed, and stop words are eliminated.
    - **Lemmatization:** Words are reduced to their root forms.
    - **Feature Extraction:**
        - TF-IDF features were used for traditional models.
        - Raw sequences (with padding) were used for the LSTM model.
        - Pre-trained embeddings and tokenization were employed for the transformer-based model.

2.2. Evaluation Metrics

The following metrics were used to assess model performance:

- **Accuracy**
- **Precision**
- **Recall**
- **F1-Score**
- **Confusion Matrix Analysis**

Best Model I choose to solve this Natural Language Inference Problem:

Transformer-Based Model (XLM-R)

Metrics:

- **Training Accuracy: 98%**
- **Evaluation Accuracy: 90%**
- **Test Accuracy: 91%**

Observations:
The model achieved the highest performance due to its deep contextual understanding of language. Although it requires more computational resources, its superior ability to model nuanced relationships in the text resulted in the best overall metrics among all tested models.

3. Model Evaluation

3.1. Baseline Model: Random Forest

- **Metrics:**
  - **Accuracy:** 44%
  - **Precision:** 0.44 (weighted average)
  - **Recall:** 0.44 (weighted average)
  - **F1-Score:** 0.44 (weighted average)
- **Observations:**
  The Random Forest model serves as a solid baseline. It performs reasonably well on this task; however, it struggles to capture the complex linguistic patterns and context between sentence pairs. Misclassifications were more frequent between the Neutral and Entailment classes.

3.2. Baseline Model: Decision Tree

- **Metrics:**
  - **Accuracy:** 40%
  - **Precision:** 0.40 (weighted average)
  - **Recall:** 0.40 (weighted average)
  - **F1-Score:** 0.40 (weighted average)

3.3. Custom Artificial Neural Network (ANN)

- **Metrics:**
  - **Accuracy: 49%**
  - **Validation Accuracy: 49%**
  - **Precision: 0.49**
  - **Recall: 0.49**
  - **F1-Score: 0.49**
- **Observations:**
  The ANN model improves on the baseline by leveraging non-linear transformations. Early stopping and dropout regularization helped prevent overfitting, and the model showed better handling of ambiguous cases compared to the Random Forest.

3.4. LSTM-Based Model

- **Metrics:**
  - **Accuracy:** 35%
  - Validation Accuracy: 36%
  - Test Accuracy-35%
- **Observations:**
  The LSTM network benefits from its ability to capture sequential dependencies in the text. This model demonstrated improved performance over the ANN, especially in

understanding the context between the premise and hypothesis, leading to fewer misclassifications.

3.5. Transformer-Based Model (XLM-R)

- **Metrics:**
  - **raining Accuracy:** 98%
  - **Evaluation Accuracy:** 90%
  - **Test Accuracy:** 91%

Observations:
The model achieved the highest performance due to its deep contextual understanding of language. Although it requires more computational resources, its superior ability to model nuanced relationships in the text resulted in the best overall metrics among all tested models.

4. Comparative Analysis

Model	Accuracy
Random Forest	44%
Custom ANN	49%
LSTM	35%
XLM-R Transformer	98%

The table above shows that as we move from traditional machine learning techniques to deep learning and finally to transformer-based models, performance consistently improves. This trend highlights the importance of contextual understanding and sequential modeling in handling complex natural language tasks.

5. Error Analysis

- **Confusion Matrix Insights:**
  - The Random Forest model exhibited higher confusion between the Neutral and Entailment classes.
  - Both the ANN and LSTM models showed improved class separation, with fewer borderline errors.
  - The BERT model achieved near-optimal class differentiation, indicating effective contextual encoding.

- **Common Errors:**
  - Ambiguities in language, especially in cases where the distinction between Neutral and Entailment is subtle, led to some misclassifications.
  - The baseline model occasionally misclassified Contradiction examples as Neutral due to overlapping vocabulary usage.

6. Conclusions and Future Work

- **Conclusions:**
  - The transformer-based XLM-R model outperformed other models, achieving an accuracy of 98% and balanced precision, recall, and F1-scores.
  - Deep learning models, particularly those that leverage sequential or contextual embeddings, are better suited for nuanced tasks like sentence pair classification.
- **Future Work:**
  - **Ensemble Methods:** Investigate combining models to further boost performance.
  - **Hyperparameter Optimization:** Explore more rigorous hyperparameter tuning to potentially enhance model performance.
  - **Data Augmentation:** Utilize techniques such as paraphrasing or back-translation to enrich the training dataset.
  - **Deployment:** Integrate the best-performing model into a real-time application or web service.

This performance evaluation report provides a comprehensive overview of the experimental setup, model performance, and insights derived from the evaluation. It serves as a crucial document to understand the strengths and limitations of each approach in the context of sentence pair classification.