IMDB Sentiment Analysis using BERT

Sai Saanvi D S – 1BG23CS128

Navya S – 1BG23CS090

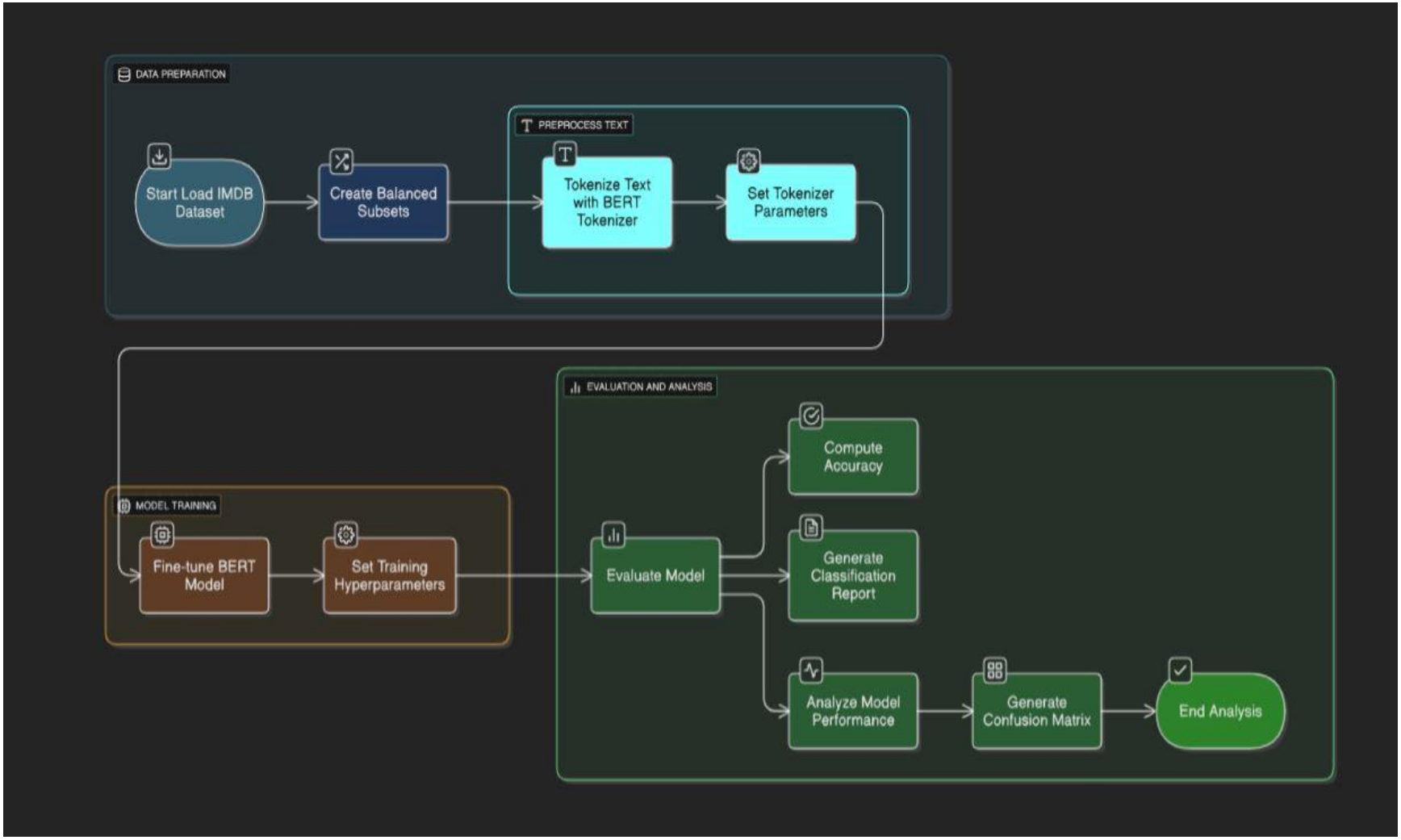
1. OBJECTIVE

- To build a deep learning model that classifies IMDB movie reviews as positive or negative.
- Utilize pre-trained BERT model for robust understanding of natural language.
- Focus on faster execution using a balanced and reduced-size dataset for experimentation.

2. METHODOLOGY

1. Load IMDB dataset from Hugging Face
2. Create a balanced subset of training and testing data
3. Preprocess the text (tokenize using BERT tokenizer)
4. Fine-tune BERT on binary classification task
5. Evaluate using accuracy and classification report
6. Analyze model performance on test data

WORKFLOW


DATA PREPARATION
Start Load IMDB Dataset
Create Balanced Subsets
PREPROCESS TEXT
Tokenize Text with BERT Tokenizer
Set Tokenizer Parameters
MODEL TRAINING
Fine-tune BERT Model
Set Training Hyperparameters
EVALUATION AND ANALYSIS
Evaluate Model
Compute Accuracy
Generate Classification Report
Analyze Model Performance
Generate Confusion Matrix
End Analysis

3. KEY ASSUMPTIONS

- Equal number of positive and negative reviews improves classification balance
- Reducing sequence length to 128 does not significantly impact accuracy
- Using a smaller training set (2000 samples) still demonstrates model behavior
- CPU training is acceptable for quick experiments

4. MODEL EVALUATION

- - Accuracy and F1 Score used for evaluation
- - Confusion matrix shows how well positive/negative reviews are predicted
- - Handles short/long reviews but may misclassify overly vague inputs
- - Sample prediction: 'The movie was amazing!' → Positive

5. SUMMARY AND OUTCOMES

- Successfully fine-tuned BERT for sentiment classification
- Achieved reasonable performance on small dataset
- Demonstrated BERT's ability to understand nuanced language in reviews
- Highlights importance of preprocessing and balanced sampling

6. FUTURE IMPROVEMENT

- Use full IMDB dataset (50,000 reviews) for better generalization
- Train on GPU for faster convergence
- Hyperparameter tuning (learning rate, epochs, max_length)
- Use more advanced BERT variants like RoBERTa or DistilBERT

7. APPENDIX

- Model: bert-base-uncased
- Libraries: PyTorch, Hugging Face Transformers & Datasets
- Dataset: IMDB movie reviews
- Train/Test Size: 2000/500
- Tokenization: BERT tokenizer with max_length = 128
- Optimizer: AdamW with learning rate = 3e-5

THANK YOU