Tweet Sentiment Classification - Model Analysis Report

PROJECT Project Overview:

This project classifies tweets into positive (1) or negative/neutral (0) sentiment using:

- XGBoost Classifier (baseline)
- Fine-tuned BERT transformer

DATASET Dataset:

- Samples: 31,962
- Train/Test: 80/20
- Features: id, tweet, label

PREPROCESSING Preprocessing:

- Lowercasing, removing mentions/hashtags/URLs
- Punctuation & stopword removal, lemmatization
- Tokenization: TF-IDF for XGBoost, BERT tokenizer for BERT

XGBOOST RESULTS XGBoost Results:

- Accuracy: 95.18%
- Precision (class 0): 0.96 | Recall: 0.99 | F1: 0.97
- Precision (class 1): 0.82 | Recall: 0.42 | F1: 0.55
- Struggles with positive class due to imbalance

BERT RESULTS BERT Results:

- Accuracy: ~97.8%
- Precision: 0.95 | Recall: 0.92 | F1: 0.93
- Handles sarcasm/context better, improved generalization

ERROR ANALYSIS Error Analysis:

- Ambiguity, sarcasm, and informal language affect performance
- BERT handles these better than XGBoost

FEATURE INSIGHTS XGBoost Feature Insights:

- Top negative words: hate, worst, angry, broke
- Top positive words: love, thanks, awesome, great

DEPLOYMENT Deployment:

- Flask API endpoint: POST /predict
- Dockerized for portability
- Streamlit UI for user input
- HuggingFace model upload (optional)

CONCLUSION Conclusion:

- BERT significantly outperforms XGBoost
- End-to-end pipeline includes training, evaluation, and deployment