Project Report

Emotion Detection from Text using Machine Learning

Intern: Irum Imran
Submission: TechnikNest AI/ML Mini Project

🔍 Problem Statement

Understanding human emotion from text is a complex challenge. It requires machines to interpret subtle cues, word patterns, and context. This project aims to build a machine learning model that can classify English sentences into different emotional categories like joy, sadness, fear, anger, etc.

🎯 Project Goals

- Train a machine learning model to detect emotions from text.
- Build a working CLI (Command Line Interface) for testing.
- Enable practical applications like sentiment monitoring, chatbot empathy, and mental health support.

📂 Dataset & Tools

- **Dataset**: HuggingFace Emotion Dataset
    - Contains labeled sentences with emotions: joy, sadness, anger, fear, surprise, love.
- **Tools & Libraries**:
    - Python, Scikit-learn, Joblib, TF-IDF
    - Google Colab (for training)
    - GitHub (for code hosting)
    - CLI (for user interaction)

🧠 Methodology

1. **Data Acquisition**:
   Pre-labeled dataset with ~20,000 emotion-tagged sentences.
2. **Preprocessing**:
   - Tokenization
   - Stopword removal
   - TF-IDF Vectorization
3. **Model Training**:
   - Trained a Logistic Regression classifier on TF-IDF vectors.
   - Saved model using Joblib for later prediction.
4. **Deployment**:
   - Built a Python CLI where users can enter sentences and get predicted emotions.

📊 Results

- **Model Accuracy**: ~85% (on test data)
- Correctly predicts clear examples like:
  - “I’m so happy today!” → joy
  - “I feel so alone.” → sadness
  - “I’m terrified of failing.” → fear

🎓 What I Learned

- Basics of NLP, including tokenization and vectorization.
- Building and evaluating machine learning classifiers.
- Model persistence using Joblib.
- Building a working CLI tool in Python.
- Documenting and managing code with GitHub.

🚀 Future Improvements

- Try deep learning models (LSTM, BERT).
- Expand emotion categories (e.g., anxiety, optimism).
- Add multilingual support.
- Turn the CLI into a web or mobile app using Streamlit or Flask.
- Deploy as an API for real-time chatbot use.

📎 Final Files

- ✅ Colab Notebook (training)
- ✅ CLI Script (cli_emotion_detector.py)
- ✅ Trained Model (emotion_model.pkl)
- ✅ TF-IDF Vectorizer (vectorizer.pkl)
- ✅ GitHub Repo
- ✅ Presentation Slides
