Project Report: Comment Categorization & Reply Assistant Tool

📁 GitHub Repository: https://github.com/donjoshy8547/comment-categorization-tool

🔗 Hosted URL: https://comment-categorization-tool-vdsxkqrdwpojcdsf59dvkn.streamlit.app/

# Project Overview

The Comment Categorization & Reply Assistant Tool is an intelligent NLP-based web application that processes user-generated comments from social media or public platforms. It classifies them into predefined categories using a hybrid machine learning model and provides automated yet empathetic response suggestions.  
  
This tool helps marketing teams, creators, and businesses efficiently understand, respond to, or flag user feedback based on its intent or emotion.

# Objectives

- Categorize user comments into meaningful emotional/intent-based classes  
- Use a hybrid ML pipeline (Logistic Regression + BERT fallback) for accurate classification  
- Build a lightweight, responsive UI using Streamlit  
- Cloud deployment via GitHub + Streamlit Cloud  
- Generate automated response suggestions per category  
- Provide basic insights using visual summaries

# Target Categories

1. Praise  
2. Support  
3. Constructive Criticism  
4. Hate/Abuse  
5. Threat  
6. Emotional  
7. Spam/Irrelevant  
8. Question/Suggestion

# Technologies Used

- Language: Python  
- Libraries: scikit-learn, pandas, nltk, transformers, Streamlit, matplotlib, joblib  
- Modeling: Logistic Regression as the primary classifier with fallback to BERT  
- UI: Streamlit  
- Deployment: GitHub, Streamlit Cloud

# Project Structure

comment-categorization-tool/  
├── app.py # Main Streamlit interface  
├── bert\_classifier.py # BERT fallback classification logic  
├── preprocess.py # Text preprocessing functions  
├── reply\_templates.py # Category-based reply generation  
├── train\_model.py # ML training pipeline  
├── models/  
│ ├── model.pkl # Trained Logistic Regression model  
│ └── vectorizer.pkl # TF-IDF Vectorizer  
├── data/  
│ └── sample\_comments.csv # Dataset used for training  
├── requirements.txt # Python dependencies  
├── README # Project overview and usage guide

# How to Run (Locally)

Model is already trained and stored inside the models/ directory.  
  
1. Clone the repository:  
 git clone https://github.com/yourusername/comment-categorization-tool.git  
 cd comment-categorization-tool  
  
2. Install dependencies:  
 pip install -r requirements.txt  
  
3. Run the app:  
 streamlit run app.py  
  
Note: nltk resources like stopwords and wordnet will auto-download if missing.

# Deployment Info

- Streamlit Cloud: Fully deployed and operational via GitHub integration  
- Hosted URL: https://comment-categorization-tool-vdsxkqrdwpojcdsf59dvkn.streamlit.app/

# Bonus Features

- Fallback logic using BERT for improving classification accuracy on ambiguous input  
- Streamlit-based bar chart visualizing category distribution  
- Automated reply generator based on detected category  
- Input via text field or file upload (CSV)

# Summary

This project effectively combines traditional ML and modern transformer techniques to deliver a robust, user-friendly feedback analyzer. It is optimized for deployment and local packaging, with a focus on real-world usability and extensibility.

Created by: Don Joshy

Phn.No : +91 9497020246

Email: donjoshy8547@gmail.com