**Phase-2 Submission Template**

**Student Name:** karthik R

**Register Number:** 421323104046

**Institution:** KRISHNASWAMY COLLEGE OF ENGINEERING AND TECHNOLOGY

**Department:** COMPUTER SCIENCE AND ENGINEERING

**Date of Submission:** 29-04-2025

**Github Repository Link: <https://github.com/sekar336/project.git>**

### **1. Problem Statement**

### *This project aims to transform customer support operations by creating an intelligent chatbot capable of handling user queries in real-time using Natural Language Processing (NLP) and machine learning techniques. By reducing human workload and improving response times, the chatbot enhances customer satisfaction and operational efficiency across multiple platforms.*

### **2. Project Objectives**

### *\*Develop an NLP-driven chatbot to automate customer query responses.*

### 

### *\*Implement intent detection and entity recognition models for high accuracy.*

### 

### *\*Integrate the chatbot into websites or apps.*

### 

### *\*Evaluate model performance through metrics like precision, recall, and F1-score.*

### 

### *\*Continuously improve the chatbot through user feedback and periodic retraining.*

### **3. Flowchart of the Project**

### NoteGPT-Flowchart-1745907193299.jpeg

### **4. Data Description**

The dataset consists of customer support tickets collected from CRM systems and public sources such as Twitter.

Data Type: Textual customer interactions.

Data Format: CSV.

Features: Ticket ID, Timestamp, Customer Query, Intent Label, Response.

Size: ~5000 records.

### **5. Data Preprocessing**

Cleaning: Removed special characters, HTML tags, and emojis.

Tokenization and Lemmatization: Used SpaCy for tokenization and lemmatization.

Stopword Removal: Eliminated common stopwords to improve model focus.

Handling Imbalanced Data: Used oversampling techniques like SMOTE where necessary.

**6. Exploratory Data Analysis (EDA)**

*Intent Distribution: Visualized using bar plots to understand query categories.*

*Common Keywords: Word clouds and frequency analysis to extract most common terms.*

*Response Times: Analyzed average customer response times.*

### **7. Feature Engineering**

*Extracted new features such as Query Length, Presence of Keywords, and Sentiment Score.*

*Performed TF-IDF vectorization for text representation.*

*Feature selection based on mutual information scores.*

### 

### **8. Model Building**

### *Intent Detection: Used Logistic Regression, Random Forest, and fine-tuned BERT models.*

### 

### *Entity Recognition: Implemented using pre-trained SpaCy NER and custom-trained models.*

### 

### *Dialogue Management: Rule-based system using Rasa framework.*

### 

### *Data Splitting: 80% Training / 20% Testing.*

### 

### *Performance Metrics:*

### 

### *Logistic Regression: Accuracy = 82%, F1-score = 0.80*

### 

### *BERT Fine-tuned: Accuracy = 91%, F1-score = 0.89*

### **9. Visualization of Results & Model Insights**

Confusion Matrix: Highlighted precision for different intents.

F1-Score Chart: Compared Logistic Regression, Random Forest, and BERT.

NER Results: Visualized extracted entities in chatbot conversations.

Key Insights:

BERT outperformed traditional models, especially in detecting complex intents. NER models successfully identified customer names, product names, and dates with 88% precision.

**10. Tools and Technologies Used**

Programming Language: Python

IDE/Notebook: Google Colab, Jupyter Notebook

Libraries: pandas, numpy, matplotlib, seaborn, scikit-learn, SpaCy, HuggingFace Transformers

Chatbot Frameworks: Rasa, Dialogflow (optional)

Deployment Tools: Streamlit, Flask

**11. Team Members and Contributions**

*Sekar: Project management, final deliverables, model tuning.*

*Srikanth: Data collection, preprocessing, and analysis.*

*Sakthi Murugan: Model development (Intent Detection and NER).*

*Senthamizh: Feature engineering and evaluation.*

*Ram Sundar: Visualization, reporting, and deployment support.*