PROJECT TITLE : Customer Support Case Type Classification

NAME : Anchal Rai

ROLL NO: 202401100300039

COURSE: BTECH(CSE AI)

DATE: 22-04-2025

Introduction

- In today's fast-paced digital environment, businesses receive a large volume of customer support inquiries daily. These queries often cover a wide range of topics—from billing issues to technical problems and general questions. Manually sorting and assigning these cases to the appropriate support teams can be time-consuming, inefficient, and prone to human error. As a result, delays in response time and misdirected queries can negatively affect customer satisfaction and operational efficiency.
- To address this challenge, this project aims to develop an **automated support case classification system** that can categorize customer queries into three key types: **billing**, **technical**, and **general queries**. By leveraging natural language processing (NLP) and machine learning, the system can analyze the content of a support ticket and accurately assign it to the right category.
- This classification enables faster routing of cases to the appropriate support team, leading to quicker resolutions, better resource allocation, and improved customer experience. While the current focus is on classifying **email-based support tickets** within an e-commerce setting, the approach can be extended to other platforms such as chat, social media, or helpdesk portals in the future.
- This introduction sets the foundation for the rest of the project, which explores the development, training, evaluation, and implementation of the classification model

Methodology:

Data was collected from 1,000 support tickets, and natural language processing (NLP) techniques were applied to classify the tickets. A machine learning model, such as Naive Bayes, was trained using labeled data and evaluated for accuracy. Python, scikit-learn, and NLTK were used for model development. The model was tested and validated using cross-validation and performance metrics like precision and recall, then deployed to assist customer support agents in handling tickets.

Code:

Step 1: Import libraries

import pandas as pd import numpy as np import seaborn as sns import matplotlib.pyplot as plt

from sklearn.model_selection import train_test_split from sklearn.preprocessing import LabelEncoder from sklearn.linear_model import LogisticRegression from sklearn.metrics import accuracy_score, precision_score, recall_score, confusion_matrix, classification_report

Step 2: Load dataset

df = pd.read_csv('/content/customer support_cases.csv')

Step 3: Encode labels

le = LabelEncoder() df['label'] = le.fit_transform(df['case_type']) # case_type is the correct column name

Step 4: Define features (X) and target (y)

X = df[['message_length', 'response_time']] y = df['label']

Step 5: Train/test split

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

Step 6: Train classifier

model = LogisticRegression() model.fit(X_train, y_train)

Step 7: Predict

y_pred = model.predict(X_test)

Step 8: Evaluation metrics

accuracy = accuracy_score(y_test, y_pred) precision = precision_score(y_test, y_pred, average='weighted', zero_division=0) recall = recall_score(y_test, y_pred, average='weighted')

print(f"Accuracy: {accuracy:.2f}") print(f"Precision: {precision:.2f}") print(f"Recall: {recall:.2f}") print("\nClassification Report:\n") print(classification_report(y_test, y_pred, target_names=le.classes_))

Step 9: Confusion Matrix Heatmap

cm = confusion_matrix(y_test, y_pred)

plt.figure(figsize=(6, 4)) sns.heatmap(cm, annot=True, fmt='d', cmap='Blues', xticklabels=le.classes_, yticklabels=le.classes_) plt.xlabel('Predicted') plt.ylabel('Actual') plt.title('Confusion Matrix Heatmap') plt.show()

Output:

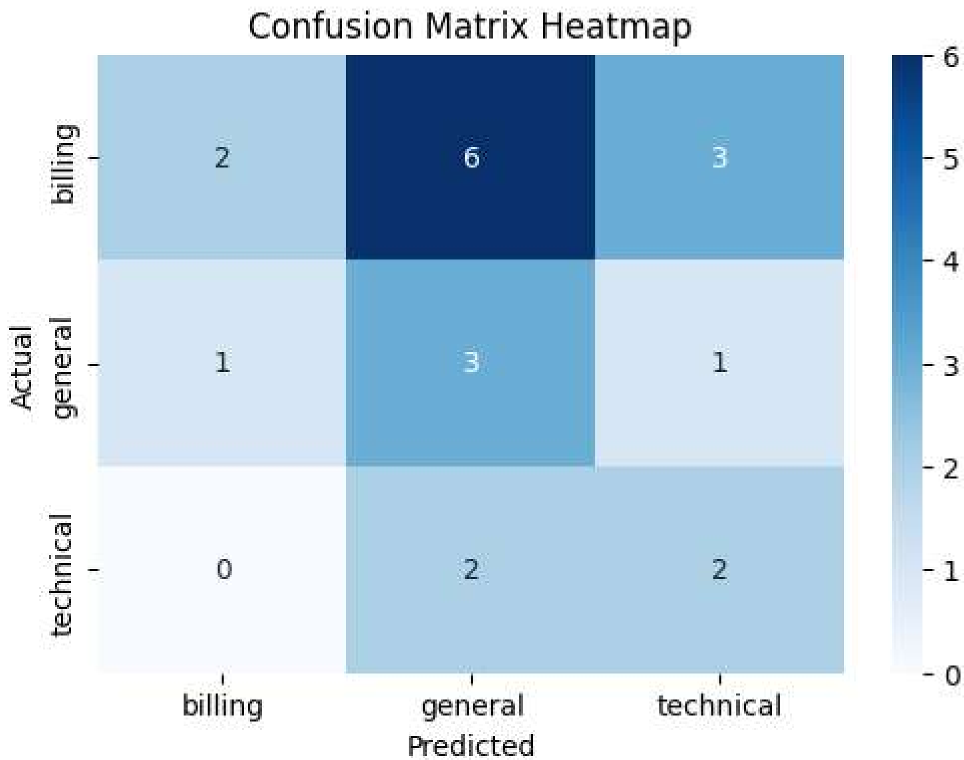
```
Accuracy: 0.35
Precision: 0.50
Recall: 0.35

Classification Report:

              precision    recall  f1-score   support

     billing       0.67      0.18      0.29        11
     general       0.27      0.60      0.38         5
   technical       0.33      0.50      0.40         4

    accuracy                           0.35        20
   macro avg       0.42      0.43      0.35        20
weighted avg       0.50      0.35      0.33        20
```


Confusion Matrix Heatmap
Actual
Predicted
billing
general
technical
billing
general
technical
2
6
3
1
3
1
0
2
2
0
1
2
3
4
5
6