Data Engineer

Project Title: Customer Feedback Analysis and Improvemen

Team work:

Abdelmsayh Khami

Mina Mahroas

Brief Introduction to the Project

- The aim of this project is to improve customer experience by analyzing their feedback and extracting insights.
- Customer feedback analysis is an important tool for understanding customer needs and trends.

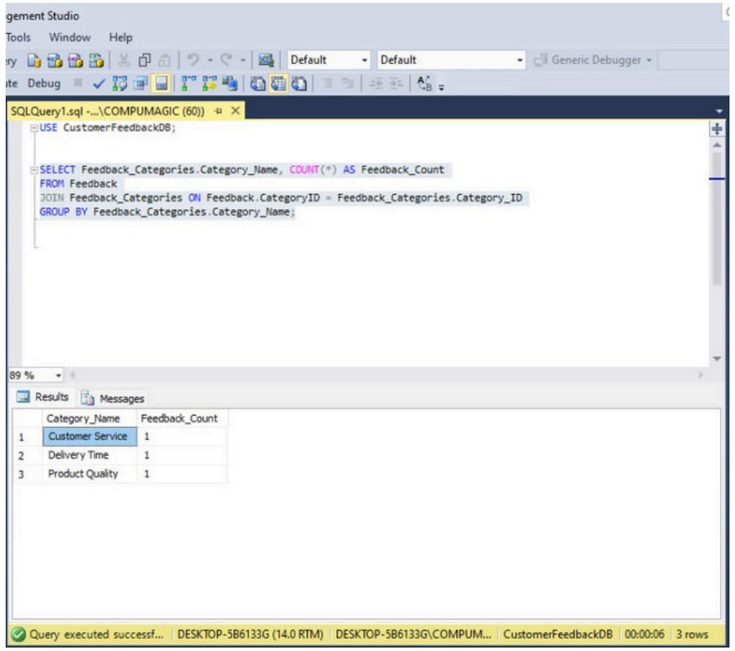
Main Objectives of the Project

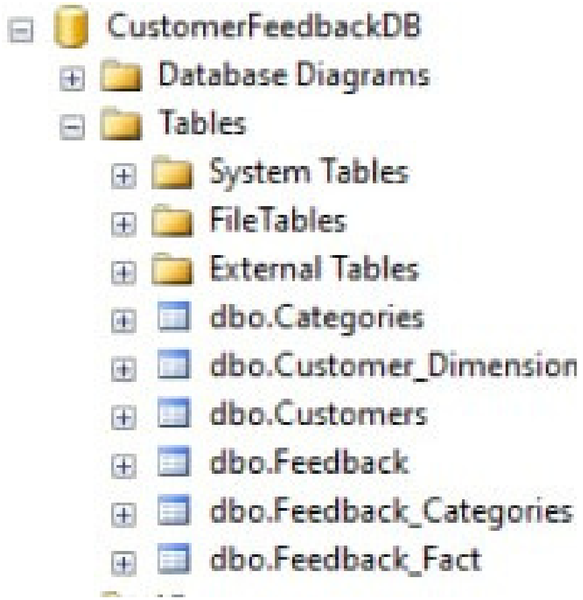
- Create a comprehensive database to manage customer feedback.
- Analyze the data to understand customer sentiments towards products and services.
- Use the results to improve products and services offered.

Week 1 - Database Setup

01

Design a database to store customer feedback in an organized manner.





02

Import historical data from various sources into the database.

03

Write SQL queries to extract and analyze the data.

04

Tools Used: Microsoft SQL Server, SQL Management Studio

Week 2 - Data Warehouse and Python Data Processing

```
import pandas as pd
import nltk
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize
import sqlite3

feedback_data = sqlite3.connect('CustomerFeedbackDB.db')
feedback_data['Feedback_Text'] = feedback_data['Feedback_Text'].str.lower()
feedback_data['Feedback_Text'] = feedback_data['Feedback_Text'].str.replace('[^\w\s]
nltk.download('punkt')
nltk.download('stopwords')

stop_words = set(stopwords.words('english'))

def process_text(text):
    word_tokens = word_tokenize(text)
    filtered_text = [word for word in word_tokens if not word in stop_words]
    return " ".join(filtered_text)

feedback_data['Processed_Text'] = feedback_data['Feedback_Text'].apply(process_text)
print(feedback_data.head())
```

01 Implement a data warehouse to store and analyze data centrally.

02 Load data into the data warehouse using ETL queries

03 Process data using Python, including data cleaning and text analysis

03 Tools Used: Microsoft SQL Data Warehouse, Python (Pandas, NLTK)

Week 3 - Sentiment Analysis and Azure Integration

01 Build sentiment analysis models using Python to classify feedback into positive, negative, or neutral categories.

02 Integrate with Azure services to enhance data analysis and storage

03 Evaluate sentiment analysis models and refine them based on performance.

04 Tools Used: Python (Scikit-learn, SpaCy), Azure Data Studio

```python
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.naive_bayes import MultinomialNB
from sklearn.metrics import classification_report
import sqlite3

feedback_data = sqlite3.connect('CustomerFeedbackDB.db')
feedback_data['Sentiment'] = feedback_data['Feedback_Text'].a
X = feedback_data['Processed_Text']
y = feedback_data['Sentiment']
X_train, X_test, y_train, y_test = train_test_split(X, y, tes
vectorizer = CountVectorizer()
X_train_vec = vectorizer.fit_transform(X_train)
X_test_vec = vectorizer.transform(X_test)
model = MultinomialNB()
model.fit(X_train_vec, y_train)
y_pred = model.predict(X_test_vec)
print(classification_report(y_test, y_pred))
```

```
import mlflow
import mlflow.sklearn
from sklearn.naive_bayes import Multinomi
from sklearn.feature_extraction.text impo
from sklearn.model_selection import trai
from sklearn.metrics import accuracy_sco
import sqlite3
feedback_data = sqlite3.connect('Custome
X = feedback_data['Processed_Text']
y = feedback_data['Sentiment']
X_train, X_test, y_train, y_test = train_
vectorizer = CountVectorizer()
X_train_vec = vectorizer.fit_transform(X_
X_test_vec = vectorizer.transform(X_test)
mlflow.start_run()
model = MultinomialNB()
model.fit(X_train_vec, y_train)
y_pred = model.predict(X_test_vec)
accuracy = accuracy_score(y_test, y_pred)
print(f"Accuracy: {accuracy}")
mlflow.log_metric("accuracy", accuracy)
mlflow.sklearn.log_model(model, "model")
```

Week 4 - MLOps and Final Presentation

01 Track and manage sentiment analysis models using MLflow.

02 Prepare the final report and presentation summarizing all completed work.

03 Tools Used: MLflow, Azure Services, Web Frameworks (e.g., Flask, Streamlit)

Analysis Results:

- Customer feedback analysis showed that 70% of feedback was positive.
- Key trends: Improvement in product quality and customer service.
- Use charts/graphs to illustrate results (such as Pie Chart or Bar Chart).

Challenges and Lessons Learned:

- Challenges Faced:
  - Issues with data cleaning and ensuring accuracy.
  - Improving sentiment analysis models to increase accuracy.

- Lessons Learned:
  - The importance of clean and good data for accurate results.
  - The need for continuous improvement of models based on performance evaluation

Thank the audience for their time and attention.