SENTIMENT ANALYSIS

OBJECTIVE:

The objective of this project is to perform sentiment analysis on a textual dataset using SQL. The dataset, sourced from Kaggle, comprises statements from diverse domains such as biodiversity conservation, climate change, social justice, space exploration, and the impact of technology. The analysis focuses on classifying the textual data into positive and negative sentiments using a keyword-based approach.

INTRODUCTION:

Sentiment analysis is a widely used technique for understanding and categorizing human emotions from textual data. This project explores a basic, keyword-based approach to sentiment analysis using SQL.

The primary goal of this beginner-level project is to classify these statements into positive and negative sentiments by identifying patterns using predefined keywords. The project employs SQL for simple data exploration and pattern-based classification, making it an accessible starting point for learning about sentiment analysis techniques.

The analysis involves:

- Exploring the distribution of sentiments within the dataset.
- Extracting subsets of data classified as positive or negative based on keywords.
- Evaluating the accuracy of the keyword-based classification.

This project provides a foundational understanding of sentiment analysis, offering a stepping stone for beginners to build upon and explore more advanced techniques such as Natural Language Processing (NLP) in the future.

TECHNOLOGY USED:

1. Database Management System

PostgreSQL: Used to store, manage, and query the dataset. Enabled efficient execution of SQL queries for data exploration, sentiment extraction, and analysis.

2. Query Language

SQL (Structured Query Language): Served as the primary tool for performing sentiment analysis. Facilitated keyword-based searches, data filtering, and classification.

3. Dataset

Kaggle Dataset: A pre-labelled dataset containing textual statements with sentiments (positive or negative). Provided a structured format for applying SQL-based methods.

SCHEMA OF THE TABLE:

Table name: **sentiment**
Attributes:

1. id (int) serves as a primary key
2. statement (varchar)
3. sentiment (text) positive or negative

ANALYSIS THROUGH SQL:

```
--Getting overall idea of the data
select * from sentiment;
select count(sentiment) from sentiment; --3309
select count(sentiment) from sentiment where sentiment='positive'; --1679
select count(sentiment) from sentiment where sentiment='negative'; --1630

--Extracting information based on some positive expressions and saved the data as "positive"
select * from sentiment where sentence like '%great%' or sentence like '%excellent%' or sentence like
'%happy%' or sentence like '%good%' or sentence like '%expected%' or sentence like '%favourite%' or
sentence like '%convenience%' or sentence like '%recommend%' or sentence like '%brilliant%' or
sentence like '%profit%' or sentence like '%hope%' or sentence like '%progress%' or sentence like
'%engagement%' or sentence like '%loved%';

--Extracting information based on some negative expressions and saved the data as "negative"
select * from sentiment where sentence like '%not%' or sentence like '%don%' or sentence like
'%sad%' or sentence like '%disappoint%' or sentence like '%bad%' or sentence like '%waste%' or
sentence like '%fraud%' or sentence like '%risk%' or sentence like '%mess%' or sentence like
'%mistake%' or sentence like '%loss%' or sentence like '%issue%' or sentence like '%threat%' or
sentence like '%barrier%' or sentence like '%misleading%';

--Getting idea about positive data
select * from positive;
select count(sentiment) from positive; --924
select count(sentiment) from positive where sentiment='positive'; --742
select count(sentiment) from positive where sentiment='negative'; --180

--Getting idea about negative data
select * from negative;
select count(sentiment) from negative; --1107
select count(sentiment) from negative where sentiment='positive'; --232
select count(sentiment) from negative where sentiment='negative'; --873
```

RESULT AND INSIGHTS:

The project processes text data by categorizing it into positive or negative sentiments using predefined keywords or a sentiment scoring system stored in SQL tables. The results are stored and queried to analyse sentiment trends, such as the percentage of each sentiment type or the most common words associated with each sentiment.

- **Percentage of each sentiment type (accuracy):**

Positive sentiments correctly classified = **80.3%**
(Correct positive sentiment rate = 742/924 ≈ 0.803)

Negative sentiments correctly classified = **78.8%**
(Correct negative sentiment rate = 873/1107 ≈ 0.788)

The classification rate is pretty much accurate.