Sentiment Analysis for Mental Health Statements

Project Purpose:

The purpose of this project was to explore and analyze the emotional patterns in mental health-related statements using Natural Language Processing (NLP), SQL, and Power BI. By examining real-world mental health data, I wanted to identify the tone and sentiment across different mental health categories such as depression, anxiety, stress, and suicidal ideation.
This project helped me practice applying machine learning techniques to text, perform structured analysis using SQL, and build an interactive visualization using Power BI to make the insights accessible and understandable.

Project Overview:

In this project, I performed an end-to-end data analytics workflow on over 52,000 mental health-related statements using Python, MySQL, and Power BI.
The objective was to analyze the emotional sentiment associated with various mental health statuses such as depression, anxiety, suicidal thoughts, and more.

Data Source:

The dataset was a combination of multiple publicly available Kaggle datasets. It included social media statements collected from platforms like Reddit and Twitter.
The dataset contained the following key columns:

- `statement` – The user's written message
- `status` – The labeled mental health category (e.g., Depression, Anxiety, Suicidal)

Step 1: Data Cleaning (Excel & Python)

I started by cleaning the dataset using Excel. I removed blank rows, dropped unnecessary columns, and used Excel formulas to trim whitespace and convert all text to lowercase. Then, in Python (Google Colab), I reloaded the dataset and repeated the cleaning steps using pandas. I dropped nulls, standardized the text with `.strip().lower()`, and reset the index.

Step 2: Sentiment Analysis in Python

To analyze sentiment, I used the TextBlob library, which returns a polarity score for each statement between -1.0 and 1.0:

- Positive sentiment: Score > 0.1
- Negative sentiment: Score < -0.1
- Neutral sentiment: Score between -0.1 and 0.1

Here's how the sentiment was calculated in Python:

```
def get_polarity(text):
    return TextBlob(text).sentiment.polarity
def get_sentiment_label(score):
    if score > 0.1:
      return 'positive'
    elif score < -0.1:
        return 'negative'
    else:
        return 'neutral'
df['sentiment_score'] = df['statement'].apply(get_polarity)
df['sentiment_label']=df['sentiment_score'].apply(get_sentiment_label)
```

These scores and labels were added to the DataFrame as new columns.

Step 3: MySQL Integration

I saved the cleaned and processed dataset to a CSV file and imported it into MySQL.
I created a table named `mental_health` with the following structure:

```
CREATE TABLE mental_health (
  id INT AUTO_INCREMENT PRIMARY KEY,
  statement TEXT,
  status VARCHAR(50),
  sentiment_score FLOAT,
  sentiment_label VARCHAR(20)
);
```

Then I loaded the data using:

```
LOAD DATA INFILE 'C:/ProgramData/MySQL/MySQL Server
8.0/Uploads/mental_health_sentiment.csv'
INTO TABLE mental_health
FIELDS TERMINATED BY ','
ENCLOSED BY '"'
LINES TERMINATED BY '\n'
IGNORE 1 ROWS
(statement, status, sentiment_score, sentiment_label);
```

Example SQL queries I used to analyze the data:

```
SELECT status, COUNT(*) FROM mental_health GROUP BY status;
SELECT sentiment_label, COUNT(*) FROM mental_health GROUP BY
sentiment_label;
SELECT status, AVG(sentiment_score) FROM mental_health GROUP BY
status;
```

Step 4: Visualization in Power BI

I connected Power BI to my MySQL database and built a dashboard that displays key insights.
The visuals I used included:

- Pie Chart – distribution of sentiment labels
- Clustered Column Chart – comparison across mental health statuses
- Table – individual statements with scores and labels
- Slicer – to filter by mental health status

Data Summary:

Total statements: 52,681
Breakdown by status:

- Anxiety: 3,841
- Normal: 16,343
- Depression: 15,404
- Suicidal: 10,652
- Stress: 2,587
- Bipolar: 2,777
- Personality Disorder: 1,077

Conclusion:

This project helped me analyze real-world mental health data and uncover meaningful patterns in the emotional tone of user statements.
Most of the data came from users labeled as either Depression, Normal, or Suicidal, making it crucial to understand how sentiment varies across these categories.
Using TextBlob, I was able to score and label over 52,000 statements.
I used MySQL to store and query the data for deeper analysis, and Power BI to build an interactive dashboard. This project demonstrates my ability to combine Python, SQL, and data visualization tools to perform comprehensive text-based analysis and generate actionable insights.