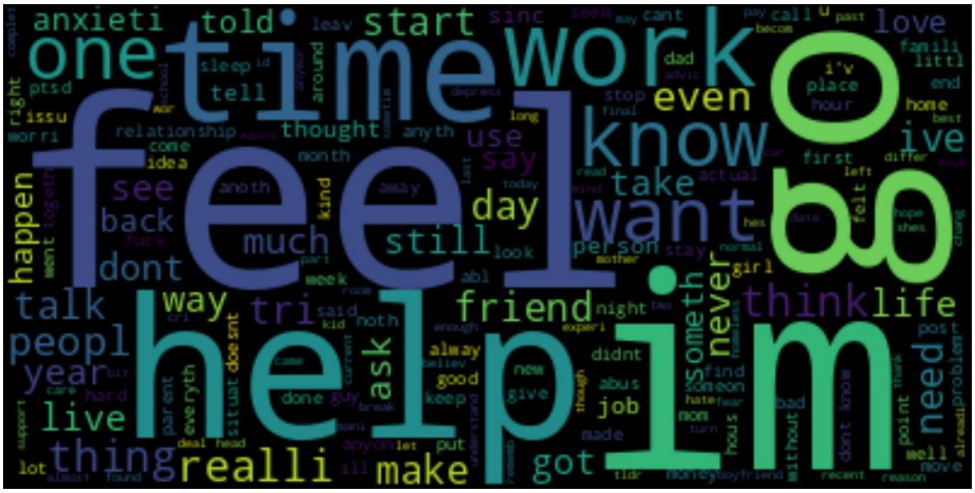
Project Topic: Stress Detection with Machine Learning

Project Overview:

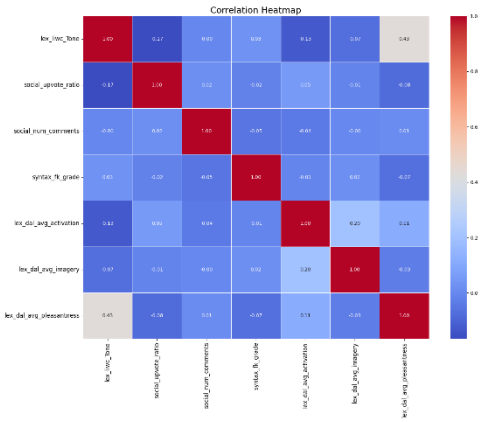
Stress, anxiety, and depression are major mental health issues affecting many people. Individuals often share their struggles on social media platforms like Instagram and Reddit. Recently, content creators and organizations have started using machine learning to detect stress in social media posts to provide timely help. This project aims to use Python to detect stress in social media posts with machine learning. The project takes a text input from the user and then predicts whether the statement shows signs of stress or not.

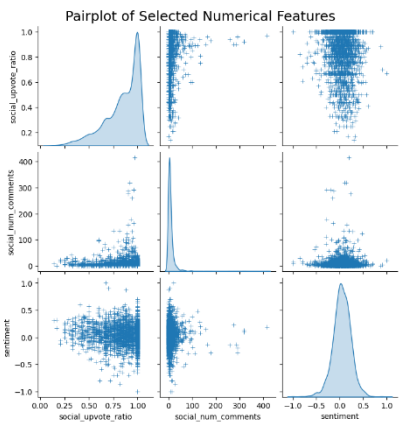
Implementation Steps:

1. **Data Collection:**
    - Dataset was collected from the internet and can be accessed here.
    - It consists of 2838 rows and 116 columns.
2. **Data Cleaning and Preprocessing:**
    - Used pandas and numpy for data transformation.
    - Cleaned text data by removing punctuations, urls, HTML tags, stopwords etc. and did stemming.
3. **Exploratory Data Analysis:**
    - Used matplotlib, seaborn and wordcloud to build visualizations of the text column.

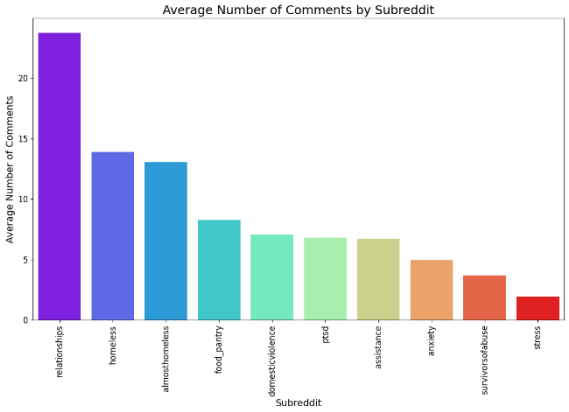


    - Created correlation heatmaps and pairplot of some selected features.





- Created a plot of Average Number of Comments by Subreddit



4. **Feature Engineering:**
   - Used tf-idf vectorization to vectorize the text data for training.
   - Did label encoding.
5. **Model Building:**
   - Implemented models like Naïve Bayes and SVM.
   - Got metrics like F1 Score, Accuracy, Precision and Recall for both models

```
user = input("Enter a Text: ")
stress = tfidf.transform([user]).toarray()
output = model.predict(stress)
print(output)
```

```
Enter a Text: Sometimes I feel like I need help
['Stress']
```

6. **Model Performance:**

Naïve Bayes

```
Accuracy: 0.7923809523809524
Precision: 0.7746031746031746
Recall: 0.8652482269503546
F1 Score: 0.8174204355108877
```

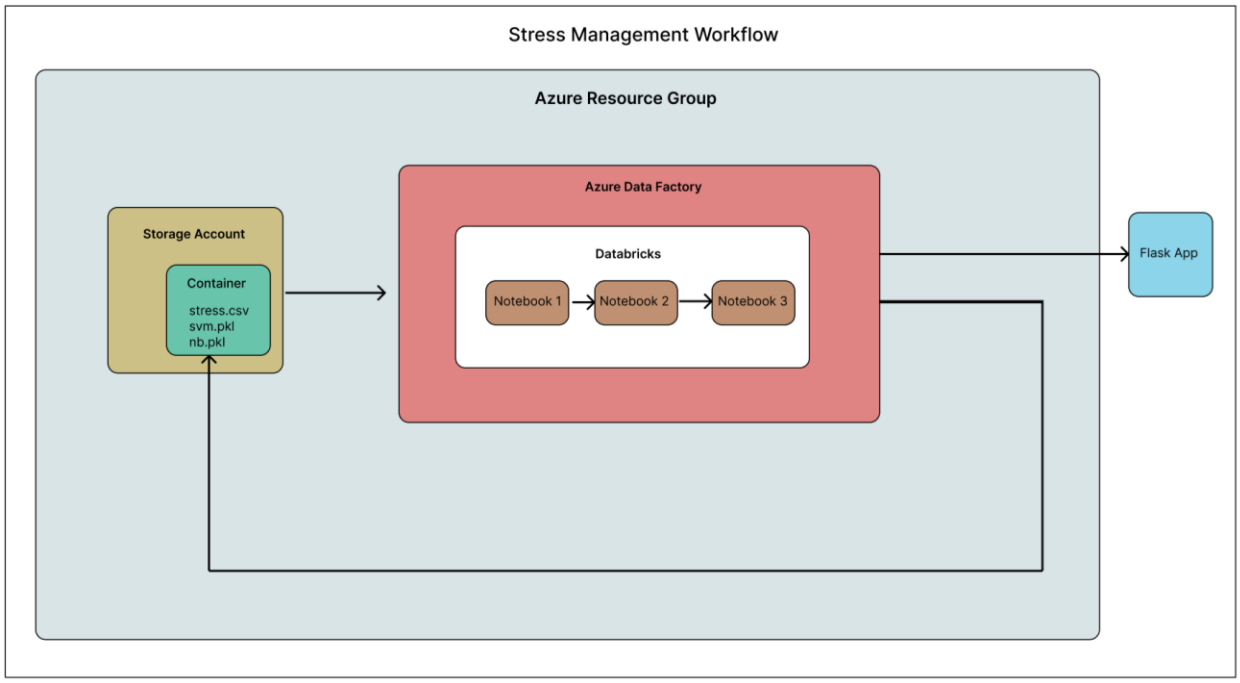
SVM

```
Accuracy: 0.7923809523809524
Precision: 0.833976833976834
Recall: 0.7659574468085106
F1 Score: 0.7985212569316082
Model saved to svc_model.pkl
```

From the above figures, it is evident that Naïve Bayes performs slightly better than Support Vector Machines (SVM). The accuracy of both models can be improved by adding more data to our dataset, enabling the models to learn the patterns more effectively. Additionally, the use of Large Language Models (LLMs) can provide a significant benefit in enhancing performance

7. **Deployment:**
   - Develop an API for real-time predictions using a Flask application.

8. **Stress management strategies:**
    - Collaborate with mental health professionals and wellness program coordinators to design and implement stress management strategies.
    - Monitor the effectiveness and adjust based on feedback.
9. **Monitoring and Maintenance:**
    - Implement automated retraining workflows using tools such as Airflow.
    - Regularly update the model with new data and monitor performance.
10. **Project Workflow**



Expected Outcomes:

- **Improved Employee Well-being**: By accurately detecting stress levels, organizations can take proactive measures to support their employees' mental health. This can lead to reduced burnout, higher job satisfaction, and overall better mental well-being.
- **Enhanced Productivity**: Identifying and managing stress can help maintain or improve productivity. Early detection allows for timely interventions, reducing the negative impact of stress on work performance and ensuring employees can work at their optimal capacity.
- **Enhanced Public Health**: On a broader scale, stress detection technology can contribute to public health initiatives by providing data on population stress levels. This can help in designing public health campaigns, allocating resources for mental health services, and developing community programs aimed at reducing stress and promoting well-being.

Tools and Technologies:

- **Programming Languages:** Python

- **Libraries:** Pandas, NumPy, Scikit-Learn, joblib , naïve_bayes, SVM, Matplotlib, Seaborn, NLTK
- **Deployment:** Flask
- **Cloud Platforms:** Azure