Twitter Sentiment Analysis Project on Real Time Scraped Dataset

1. Project Overview

This project focuses on analyzing sentiments from Twitter data using a combination of Natural Language Processing (NLP) techniques and machine learning models. The aim is to classify the sentiments of tweets into categories such as positive, negative, or neutral. This can be useful in gauging public opinion on specific topics or events.

2. Key Components

- **Data Collection**: The project utilizes the snscrape module to gather tweets based on a specific search query. Tweets are collected using various filters and are stored for further analysis.
- **Data Preprocessing**: The collected tweets undergo several preprocessing steps to clean and prepare the data for sentiment classification. This includes:
    - Removal of stopwords.
    - Tokenization of words.
    - Stemming of tokens using the PorterStemmer.
- **Feature Extraction**: Using methods like word_tokenize, the processed text data is transformed into numerical features for model training.

3. Model Training and Evaluation

- The model is trained using a sentiment classification algorithm, which classifies tweets into various sentiment categories. Several models, including Support Vector Machines (SVM), Random Forest, and Naive Bayes, are used to predict sentiments.
- **Training and Validation**: The training process includes splitting the data into training and testing sets. The model's accuracy and loss are monitored, and hyperparameter tuning is done to improve performance.

4. Prediction and Model Saving

- After training, the model is saved and deployed for future use. Predictions can be made on new, unseen data by loading the trained model.
- **Model Deployment**: A .py script is provided to load the saved model and make predictions based on new Twitter data.

5. Limitations

- **Data Imbalance**: One of the significant limitations of this project is the imbalanced dataset. The number of tweets in each sentiment category is not evenly distributed, which may cause the model to perform poorly on underrepresented classes.

    - **Recommendation**: A potential solution to this is using techniques like Synthetic Minority Over-sampling (SMOTE) or adjusting class weights in the training process to handle imbalanced data.

- **Overfitting**: The model may face issues with overfitting if the training set contains noisy or irrelevant data. Cross-validation and regularization techniques could be employed to mitigate overfitting.
- **Limited Dataset**: The model's performance is highly dependent on the size and diversity of the training dataset. A limited number of tweets can affect the model's generalization to new data.

6. Future Improvements

- Implementing advanced NLP techniques such as Transformer-based models (BERT or GPT) to improve the classification accuracy.
- Addressing data imbalance by generating synthetic samples for the underrepresented sentiment categories.
- Incorporating additional features such as tweet metadata (e.g., user location, time of tweet) to enrich the model.

7. Conclusion

This project provides a foundational approach to sentiment analysis using Twitter data. With future enhancements, the accuracy and robustness of the model can be significantly improved, making it suitable for real-world applications.