

Project - 3

Data Science Project: Sentiment Analysis

Project Title: Sentiment Analysis

Dataset:

<https://www.kaggle.com/datasets/abhi8923shriv/sentiment-analysis-dataset/data>

Project Overview:

Sentiment Analysis is a data science project that involves the use of machine learning techniques to analyze and classify textual data based on the sentiment expressed. The project aims to build a predictive model capable of determining whether a given text conveys positive, negative, or neutral sentiment.

Project Objectives:

1. Data Exploration:

- Explore the Sentiment Analysis dataset to understand its structure, features, and size.
- Identify key variables such as text content and sentiment labels.

2. Data Preprocessing:

- Perform text preprocessing tasks, including lowercasing, removing stop words, and handling special characters.
- Tokenize and lemmatize words to prepare the text for sentiment analysis.

3. Exploratory Data Analysis (EDA):

- Conduct exploratory data analysis to gain insights into the distribution of sentiment labels.
- Visualize the distribution using histograms or pie charts to understand the balance of sentiment classes.

4. Text Vectorization:

- Convert the preprocessed text into numerical vectors using techniques like TF-IDF (Term Frequency-Inverse Document Frequency) or word embeddings.
- Choose an appropriate vectorization method based on the characteristics of the dataset.

5. Model Selection:

- Explore and implement different machine learning models suitable for text classification, such as Naive Bayes, Support Vector Machines, or deep learning models like LSTM (Long Short-Term Memory) networks.
- Evaluate the performance of each model using metrics like accuracy, precision, recall, and F1 score.

6. Hyperparameter Tuning:

- Fine-tune the hyperparameters of the selected model to optimize its performance.
- Utilize techniques like grid search or random search for hyperparameter optimization.

7. Cross-Validation:

- Implement cross-validation techniques to assess the generalization performance of the model and prevent overfitting.

8. Model Interpretability:

- Interpret the model's predictions by analyzing feature importance or using techniques like LIME (Local Interpretable Model-agnostic Explanations).
- Understand which words or features contribute most to sentiment predictions.

9. Evaluation Metrics:

- Evaluate the model's performance using relevant evaluation metrics for sentiment analysis, such as confusion matrix, precision-recall curves, and ROC-AUC.

10. Deployment (Optional):

- Deploy the trained model for real-time sentiment analysis, creating an API or integrating it into a web application.
- Showcase the model's functionality in a user-friendly interface.

11. Documentation:

- Create comprehensive documentation covering data preprocessing steps, model development, and evaluation results.
- Include code snippets, visualizations, and explanations to aid understanding.

Submission Instructions:

1. Follow all the instructions mentioned in the attached Instruction PDF.
2. Zip the entire project file, including all necessary documents, assets, and source code.
3. Submit the zipped file through Google Classroom.
4. Include the GitHub repository link (if applicable) and the live website link in the submission form.

We encourage you to approach this project with creativity and professionalism. Success in this project will further enhance your internship profile. Ensure timely project submission.