**Sentiment Analysis with Basic ML Models**

# Created by Kaushik Kar

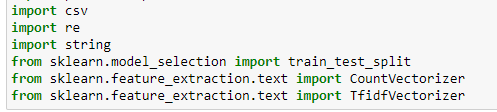
# Employment id- 2216027

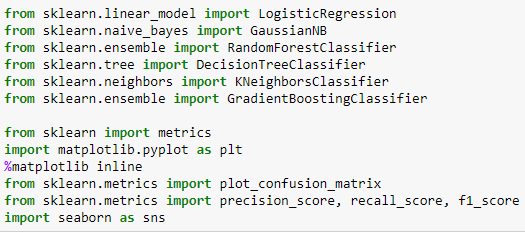
# Importing Libraries:

First, importing the important external Python packages using the pip package manager.

Graphical user interface, text

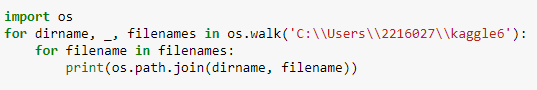
Description automatically generated





1. NumPy is used for mathematical operations like addition, subtraction, multiplication, division, etc. on arrays and matrices.
2. Pandas provides data structures for efficiently storing and manipulating large datasets, and tools for reading and writing data to and from various file formats, including CSV, Excel, and SQL databases
3. Seaborn is a data visualization library based on Matplotlib which is a plotting library used for creating static, interactive, and animated visualizations in Python.
4. torch is a PyTorch library for creating and training neural networks.
5. bs4 (Beautiful Soup) is a Python library for web scraping and parsing HTML and XML documents.
6. string is a built-in Python library that provides a collection of constants and functions for working with strings.
7. sklearn (Scikit-learn) is a popular Python library for machine learning and data analysis.
8. transformers is a library developed by Hugging Face that provides state-of-the-art pre-trained models for natural language processing tasks.
9. In particular, it seems that you are using the AutoModelForSequenceClassification and AutoTokenizer classes from the transformers library to build a sequence classification model, and the Trainer and TrainingArguments classes to train the model. The model\_selection and metrics modules from sklearn might be used for evaluating the performance of the trained mode.
10. LogisticRegression, GaussianNB, RandomForestClassifier, DecisionTreeClassifier, KNeighborsClassifier, and GradientBoostingClassifier are all different classifiers that can be used to build machine learning models for classification tasks.
11. metrics, plot\_confusion\_matrix, precision\_score, recall\_score, and f1\_score are all evaluation metrics that can be used to assess the performance of these classifiers.

This below code is using the OS module to walk through a directory tree and print out the path of each file.



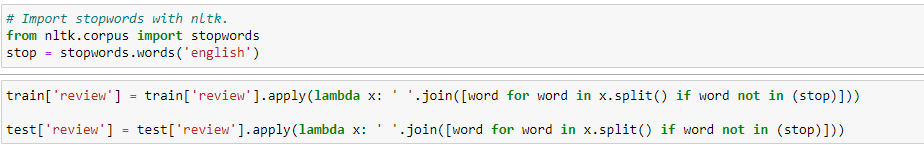
**2.Data Import**

With the help of Pandas library, we can read and upload the data in csv form.

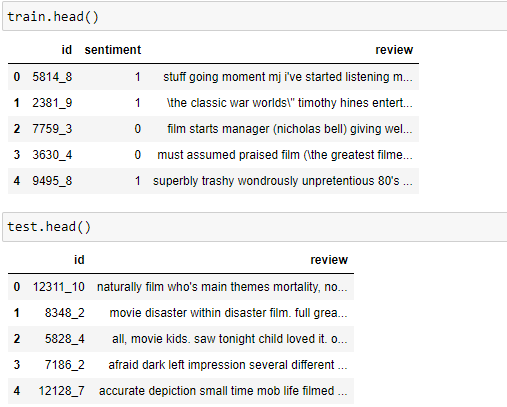


1. **Preprocessing the data**

Preprocesses text data in the 'review' column of a Pandas DataFrame named 'train'. The code removes hashtags, mentions, and URLs from the text using regular expressions and converts all the text to lowercase. These preprocessing steps are often done in natural language processing and machine learning tasks to standardize the text and remove noise, making it easier to analyze and build models.

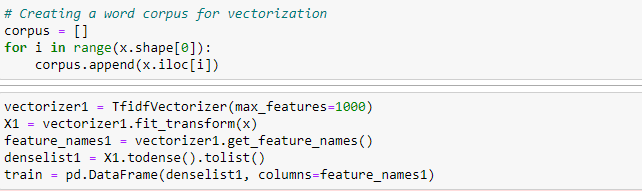


Removing stop words can be useful in natural language processing and machine learning tasks because they are often common and do not add much meaning to the text. By removing them, we can reduce the dimensionality of the text data and improve the accuracy of models.



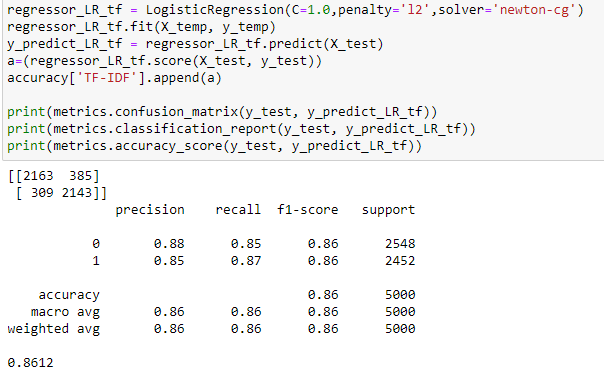
1. **Vectorization TF-IDF:**

Creating a word corpus for vectorization. The resulting 'train' DataFrame contains a bag-of-words representation of the 'x' data, where each document is represented as a vector of TF-IDF weights for the 1000 most important features (i.e., words). This type of representation is often used in natural language processing and machine learning tasks as a way to standardize the text data and prepare it for modeling.



**Testing With Models:**

**Logistic Regression:** Logistic regression is a statistical method used for binary classification, which involves predicting a binary outcome (e.g., yes or no, true or false) based on one or more predictor variables.

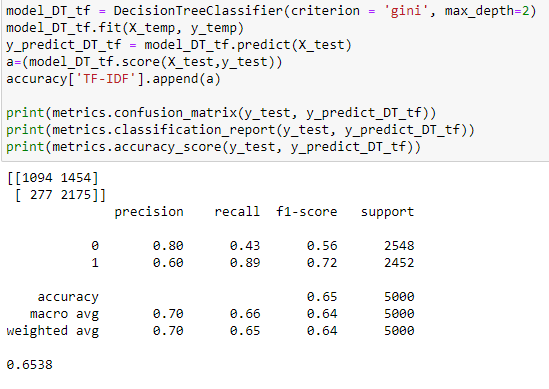


The confusion matrix shows that the classifier predicted 2163 true negatives and 2143 true positives, as well as 385 false negatives and 309 false positives. This means that out of the total 5000 test samples, 2163 negative reviews were correctly predicted as negative, and 2143 positive reviews were correctly predicted as positive.

The classification report shows that the classifier has a precision of 0.85 for predicting positive reviews, meaning that out of all the samples predicted as positive, 85% were actually positive. The recall for positive reviews is 0.87, meaning that out of all the actual positive reviews, 87% were correctly predicted as positive. The f1-score is 0.86, which is the harmonic mean of precision and recall.

The overall accuracy of the classifier is 0.8612, which means that it correctly predicted the class label for 86.12% of the test samples.

**Decision Tree Classifier**: Decision Tree Classifier is a popular supervised machine learning algorithm used for classification and regression tasks. In decision tree classifier, the dataset is split into smaller subsets using a set of decision rules based on the features of the data.

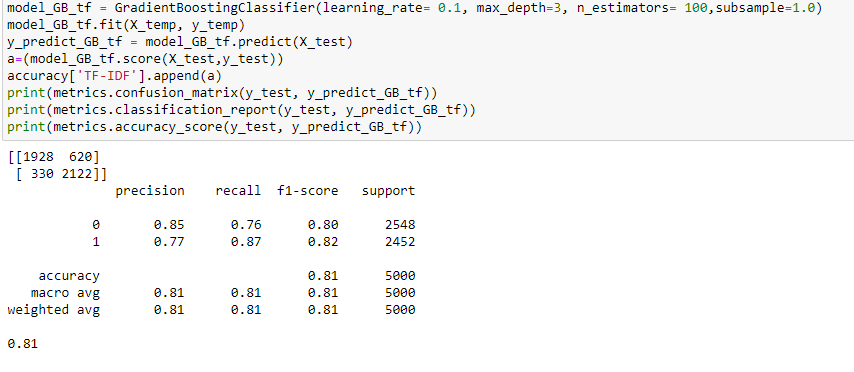


The confusion matrix shows that the classifier predicted 1094 true negatives and 2175 true positives, as well as 1454 false negatives and 277 false positives. This means that out of the total 5000 test samples, 1094 negative reviews were correctly predicted as negative, and 2175 positive reviews were correctly predicted as positive.

The classification report shows that the classifier has a precision of 0.6 for predicting positive reviews, meaning that out of all the samples predicted as positive, 60% were actually positive. The recall for positive reviews is 0.89, meaning that out of all the actual positive reviews, 89% were correctly predicted as positive. The f1-score is 0.72, which is the harmonic mean of precision and recall.

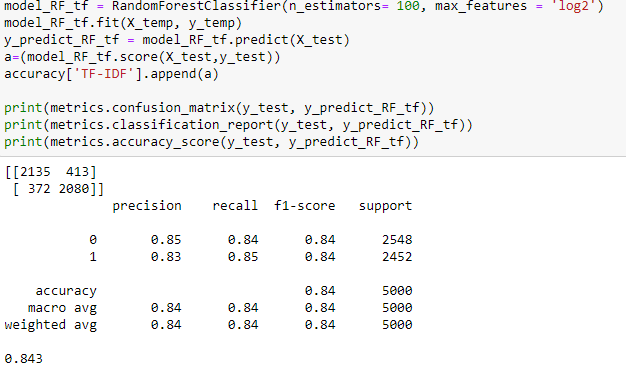
The overall accuracy of the classifier is 0.6538, which means that it correctly predicted the class label for 65.38% of the test samples. The performance of the decision tree classifier seems to be poorer than that of the logistic regression classifier you tried before.

**Gradient Boosting:** Gradient boosting is another machine learning algorithm that can be used for classification tasks. It is an ensemble method that combines multiple weak learners (usually decision trees) to create a strong learner.

****

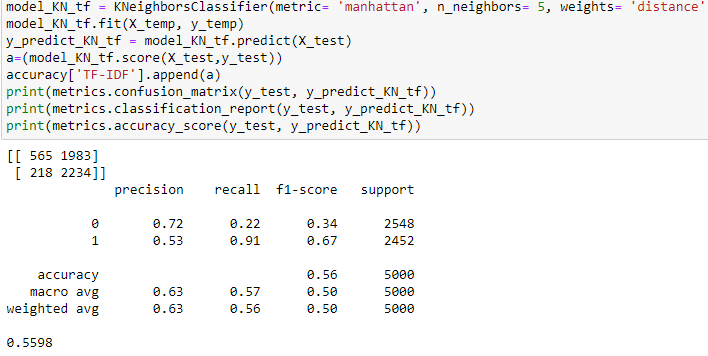
Gradient boosting is a powerful algorithm for classification tasks and can often outperform other algorithms such as logistic regression or decision trees. obtained an accuracy of 0.81 on the test set.

**Random Forest:** Random Forest is a popular ensemble learning technique used in machine learning for classification, regression, and other tasks. It is a collection of decision trees, where each tree is built on a randomly sampled subset of the training data and a randomly sampled subset of the input features.



The results show that the Random Forest algorithm achieved the highest accuracy of 0.843 on the test set, followed by Logistic Regression with an accuracy of 0.8612, Gradient Boosting with an accuracy of 0.81, and Decision Tree with the lowest accuracy of 0.6538.

**k-nearest neighbour:** The k-nearest neighbors algorithm, also known as KNN or k-NN, is a non-parametric, supervised learning classifier, which uses proximity to make classifications or predictions about the grouping of an individual data point.

****

The results from the k-Nearest Neighbors (KNN) model using TF-IDF features show a lower accuracy score compared to the other models. The model correctly classified 56% of the reviews in the test set. The confusion matrix shows that the model predicted 1983 negative reviews as positive, and 218 positive reviews as negative. The classification report also shows a lower precision, recall, and F1-score for the negative class compared to the positive class. This indicates that the model had more difficulty correctly identifying negative reviews compared to positive reviews. Overall, the KNN model using TF-IDF features performed worse than the other models in this experiment**.**