

**ADVANCED DATABASE MANAGEMENT**

**Fall 22 CPSC 531-03**

**Project Report**

Prepared by

Vaibhav Monpara

(885198523)

([vaibhavmonpara@csu.fullerton.edu](mailto:vaibhavmonpara@csu.fullerton.edu))

Vasukumar Lakhani

(885226365)

([vasulakhani@csu.fullerton.edu](mailto:vasulakhani@csu.fullerton.edu))

Nishantkumar Asodariya

(885226365)

(nishantasodariya[@csu.fullerton.edu](mailto:vasulakhani@csu.fullerton.edu))

**Outline**

1. Introduction
2. Functionalities
3. Architectures and Design
4. Github location and code
5. Steps to run the application
6. Test Results

**Introduction**

The most crucial factor in any type of business is the customer's contentment and their feedback. Every time we use Uber, we are prompted to rate both the ride itself and the performance of the driver. The significance of these ratings to the company is well understood. Reviews and ratings are a terrific technique to raise the caliber of their drivers and build a solid reputation for the business. Additionally, they get the chance to identify their weaknesses. With the aid of Bag-of-words, we have begun to classify the text reviews that Uber users have provided as feedback through this project.

As we have used the technique which is totally automated, emotional analysis was utilized here to save time and effort while providing a high-performing answer. It pulls critical information about emotions and attitudes, allowing it to understand how clients felt about their experience. When trained with more data, the analysis becomes more accurate and intelligent. Drawbacks: Unigrams were employed in the code for analysis. Using bigrams or trigrams would have resulted in more precise findings.

**Functionalities**

1. Data cleaning:

* Cleaning up the data is the first and most crucial duty. Data cleaning guarantees improved prediction and accuracy.
* Starting with the rows with ride rating = 3, we remove them. With a compromised rating, accuracy decreases rapidly and serves no useful purpose for us.
* The next step is to compile a list of frequently used words, remove them from the review column, and save them in a new column for later use.
* Tokenization is also used on the cleaned column. Each word in the text receives a token as a result.

2. Count Vectorization:

* We get a tokenization through a token and then we apply count vectorization and we get all numbers for all the token. So we can easily analyze the review.

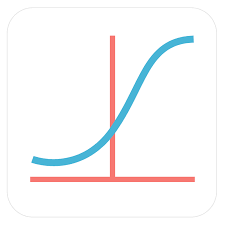
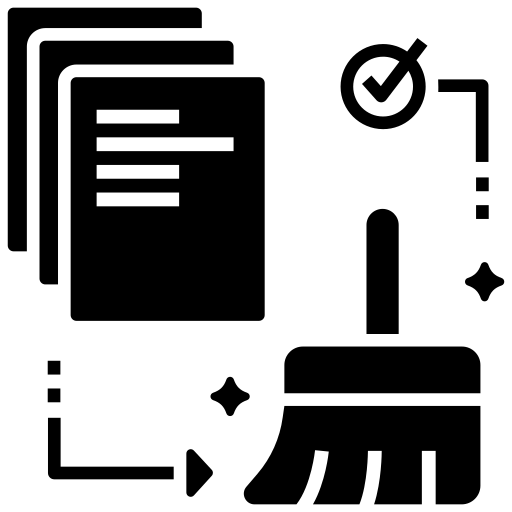
3. Logistic Regression:

* We first divide the data into Training data and Testing data before using any model.
* Typically, training data make up 70% of the data and testing data make up 30%.
* The model is trained using the training data, and it is then tested using the testing data.
* By means of the CountVectorizer function, we declare the characteristics.
* We use the training data to fit the Logistic Regression model.After the data has been trained, we put it to the test by feeding it the testing data.

4. Graph Plotting

* Here we have used a logistic regression model to plot the graph wherein we took the x-axis as sentiment and y-axis as beta coefficient. This graph shows the train and test data.

**Architectures and Design**



Dataset

Tokenization and Countvectorizer

Output from LR model

**Github location and code**

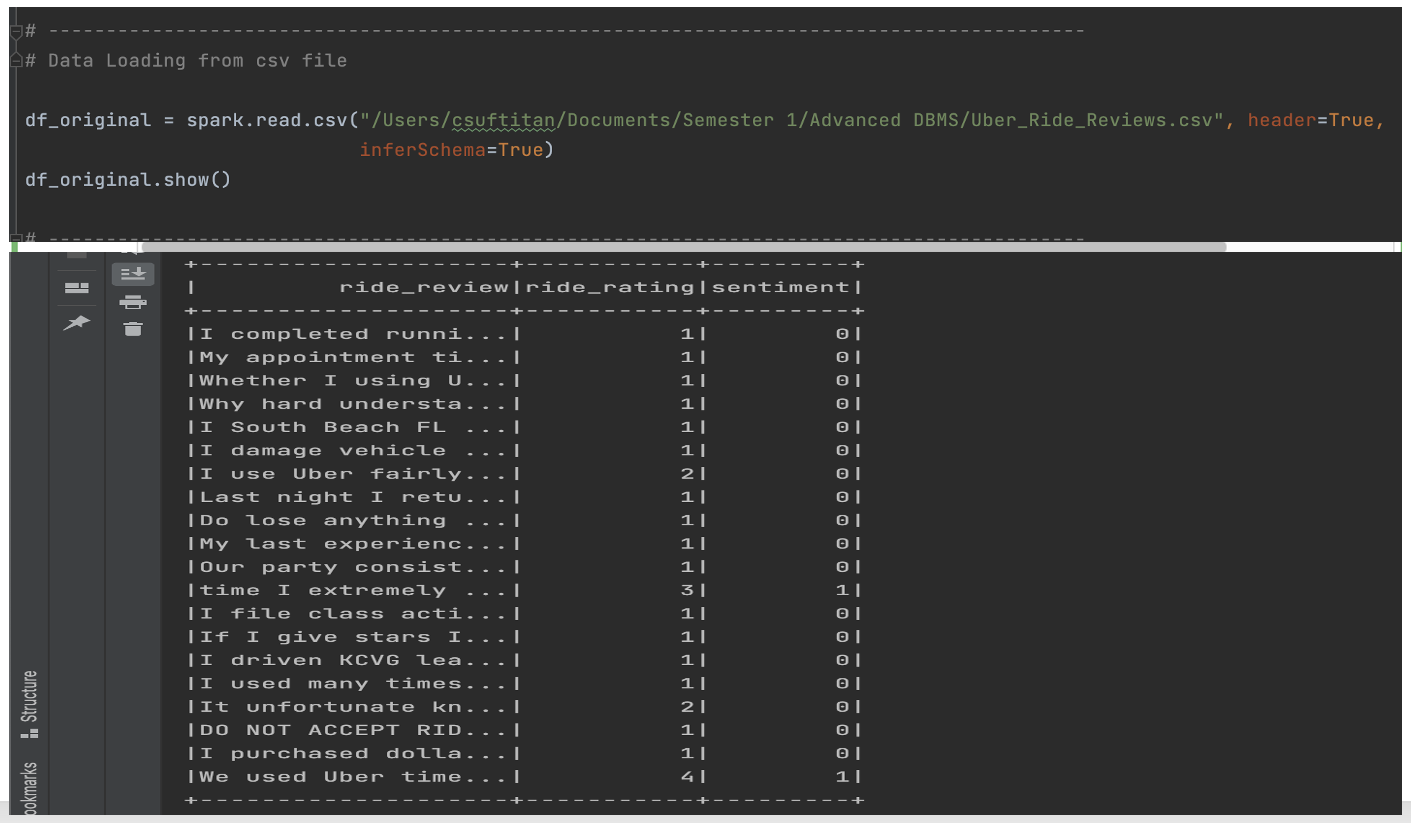
<https://github.com/VaibhavMonpara/ADV-DBM-PROJECT>

**Steps to run the application**

* Cloning the repository or directly downloading the py file from the code.
* Python version 2.8.0 or above and install packages such as pandas, numpy, matplotlib, pyspark if not present in the system.
* Output graphs will be generated about the logistic regression along with the prediction of the reviews.
* Data regarding the analysis will also be shown to the user.

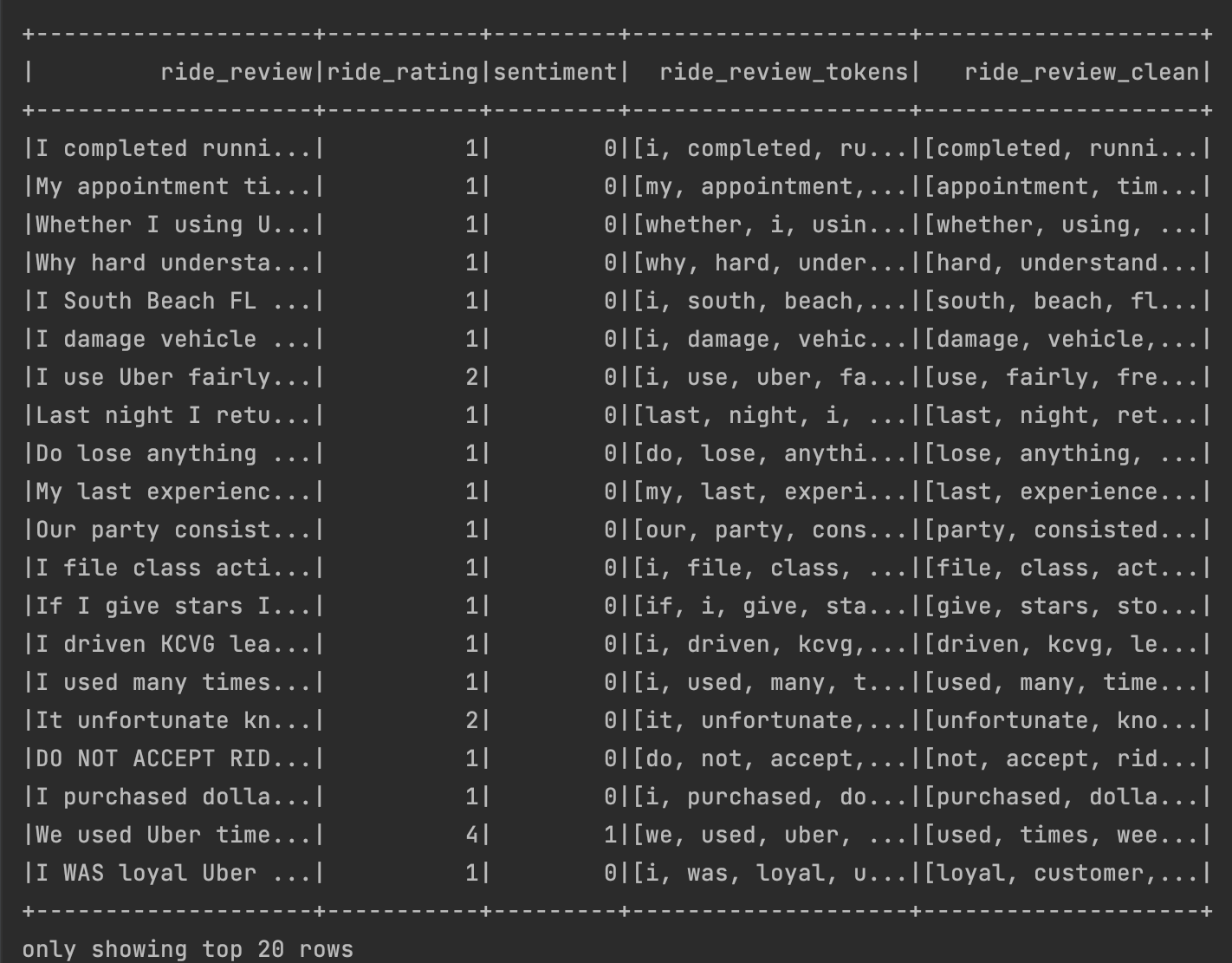
**Test Results**

1.Output of dataset and its source code

****

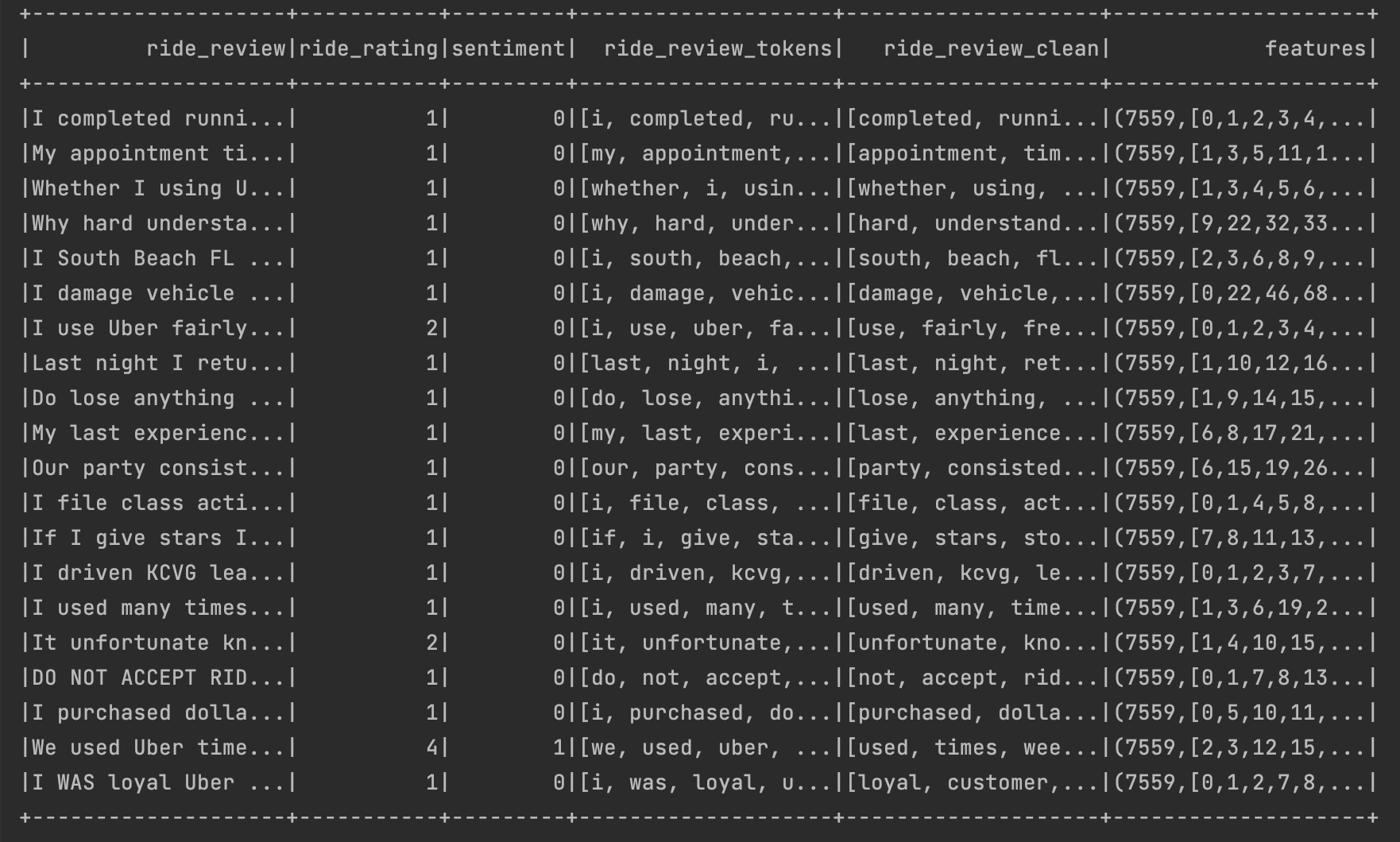
*Figure 1: Output of the dataset and its source code.*

2. Output of data cleaning process

**

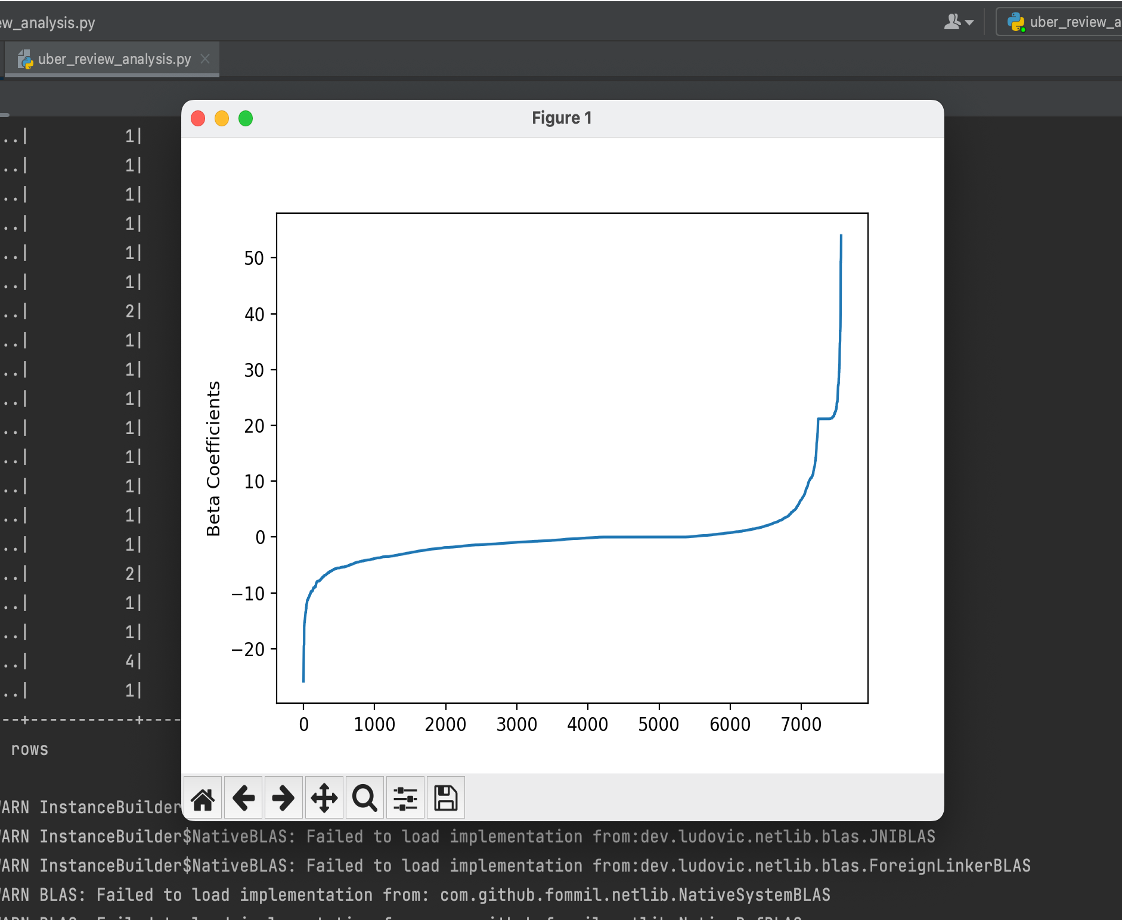
*Figure 2: Output of data cleaning process*

3. Output of clean data and feature generated from the count vectorizer function

**

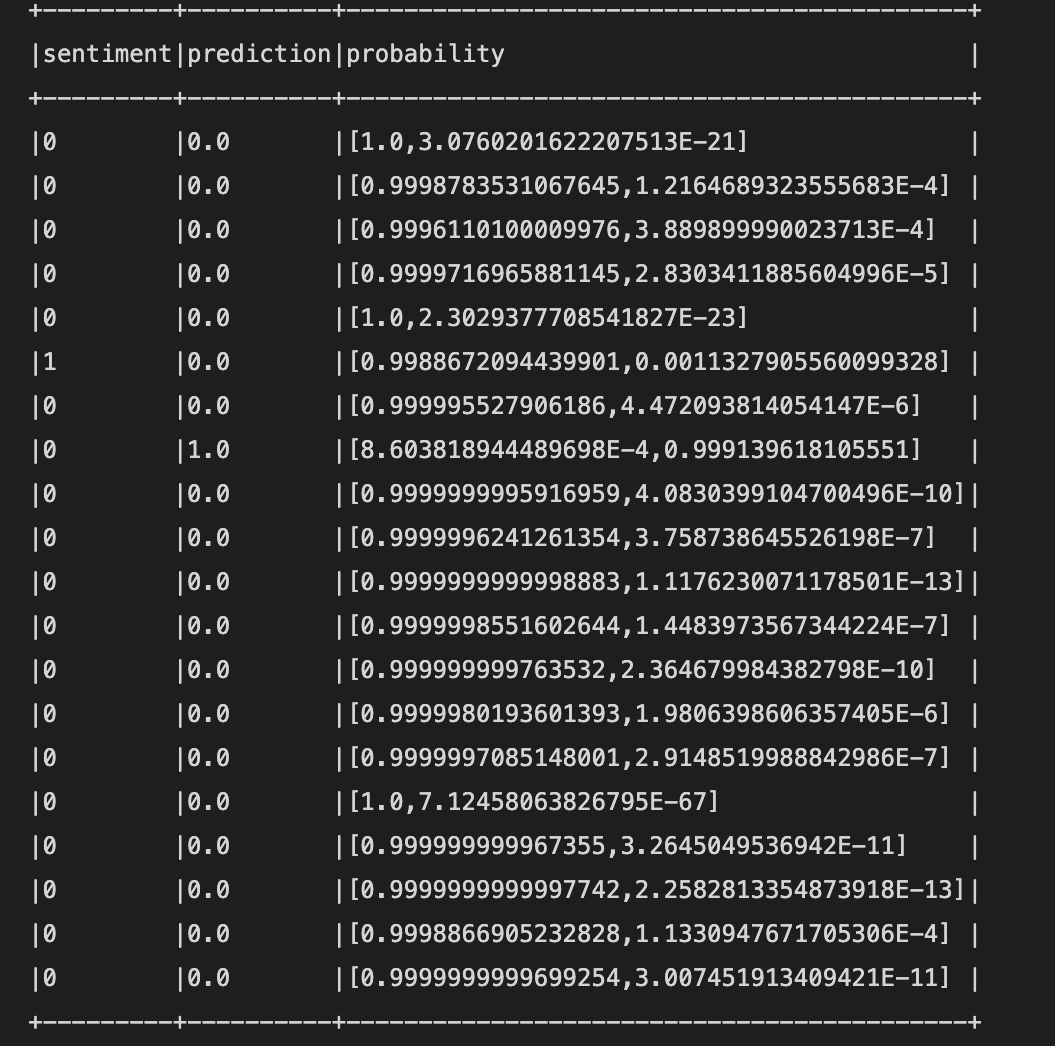
*Figure 3: Output of clean data features generated from the count vectorizer*

4. Output of logistic regression graph

**

*Figure 4: Output of the logistic regression graph*

5. Output for sentiment and prediction values

**

*Figure 5: Output of sentiment and prediction values*