Data Mining Project 1: Sentiment Analysis of Baby Product Reviews

Lalithanjana Kollipara (G01386376)
Graduate Student, Computer Science
Mail-id: lkollipa@gmu.edu
Miner Username: nocturnal
Rank: 177
Accuracy Score: 0.87

AIM:

To implement a Logistic Regression classifier model to predict whether the review is a positive review or a negative review with the help of the Bag of Words text representation method, with and without TF-IDF Vectorization.

RESOURCES PROVIDED:

1. The **train_file.dat** file consists of 18506 Baby Product reviews along with the sentiment analysis for each review. A +1 represents that the review is positive and a -1 represents that the review is negative.
2. The **test.dat** file consists of an additional 18506 Baby Product reviews but without any sentiment analysis.
3. The **format.dat** file consists of the format in which the sentiment analysis for each of the reviews in test.dat file should be saved.

CONCEPTS USED:

1. **Natural Language Processing:** Natural Language Processing is a branch in Artificial Intelligence that is used by the computer to analyse, understand and make meaningful observations on the human language in a smart and useful way. Using NLP, we can organize and structure the data to train the models and make predictions on the test data.
2. **Logistic Regression:** Logistic Regression is an example of supervised learning. Logistic Regression is a statistical analysis technique that predicts the outcome, either yes or no, based on the prior observations made on the data. It predicts the value of a dependent variable by observing the relationship between one or more existing independent variables. Logistic Regression is easy to implement and works on linear and non-linear relationships and different data-types.
3. **Bag of Words:** Bag of Words is a text representation technique used in natural language processing to convert large files consisting of huge text data into a bag of words. It doesn't any grammar or punctuation into consideration while transforming the data. It simply maintains a word count for each and every word in the text data file.
4. **Count Vectorization:** Count Vectorization is one of the methods to implement the Bag of Words concept. It breaks down the sentences or any large text files into words. It is used to convert the data files into vectors on the basis of the frequency of each word the occurs in the text data file.
5. **TF-IDF:** Term Frequency - Inverse Document Frequency is a statistical measure which evaluates the importance of a word in the documents. It calculates two different values - Term Frequency which determines the frequency of each word in the document and Inverse Document Frequency which determines the importance of the word in the text document.

IMPLEMENTATION:

1. **Data Pre-Processing:** Data Pre-Processing is done to remove the unnecessary data from the training data file. There are a total of 18506 entries in the training data file and each entry contains a review and it's sentiment analysis number.
   a. **Division of File into Columns:** The file is converted to a data frame (train_data) which contains two columns - Point (sentiment analysis number) and Review (review of the Baby Product).

b. **Handling Missing Values:** The exploratory study performed on the train data file has shown that there are a few missing values in the Review column of the train_data data frame. The missing values' data is as follows:

Index	Point	Review
3710	1	NaN
4959	-1	NaN
8199	1	NaN
8328	1	NaN
10801	1	NaN
12542	-1	NaN
16970	-1	NaN
17636	-1	NaN
18119	-1	NaN

c. **Converting to Lower Case:** All the entries in the Review column of the train_data data frame were converted into lower case alphabets for better readability of the data.

d. **Removing all the Special Characters, Punctuations and Numbers:** All the special characters, punctuations and numbers from the entries in the Review column are removed as they are unnecessary.

e. **Lemmatization:** We lemmatize each and every word in each entry of the Review column to convert that into its root word form for easy training of the model.

f. **Stop Words Removal:** Stop words were also removed to prevent model from training noisy data.

2. **Splitting of data into Train and Test data:** 75% of the data in train_data frame is being used for training the model and 25% of the data in train_data frame is being used as validation to check the accuracy of the prediction model.
3. **Data Modelling:**

   a. **Count Vectorization:** Bag of Words technique was implemented using the CountVectorizer() from the python sklearn library. Best Accuracy observed with the below parameters:

```
count_vec = CountVectorizer(min_df = 10, token_pattern = r'[a-zA-Z]+')
```

   b. **TF-IDF Vectorization:** TF-IDF technique was implemented using the TfidfVectorizer() from the python sklearn library. Best Accuracy observed with the below parameters:

```
tfidf_vec = TfidfVectorizer(min_df = 10, token_pattern = r'[a-zA-Z]+')
```

   c. **Logistic Regression:** Logistic Regression was implemented using the LogisticRegression() from the python sklearn library.

4. **Accuracy Score:**

   a. LogisticRegression with CountVectorizer = 0.8608

   b. LogisticRegression with TfidfVectorizer = 0.8703

5. **Generating the .csv file:** The trained model is then tested on the test data file containing 18506 reviews. A .csv file is generated with each of 18506 reviews' sentiment analysis.

<u>RESULTS:</u>

1. The Logistic Regression Classifier has an accuracy of **87.03%** with TF-IDF Vectorization on the training data.
2. The Logistic Regression Classifier has an accuracy of **86.08%** with the Count Vectorization on the training data.