

**CIS 593 – SPECIAL TOPICS IN CIS**

**BIG DATA**

**FINAL PROJECT REPORT**

on

**TWITTER SENTIMENT ANALYSIS on INDIA’s DEMONETIZATION EVENT**

by

**DIVIJ JOBANPUTRA**

**CSU ID: 2819861**

**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
| **Sr. No.** | **CONTENT** | **Pg. No.** |
|  |  |  |
| 1 | ABSTRACT | 3 |
| 2 | INTRODUCTION | 4 |
| 3 | STEPS OF APPLICATION | 5 |
| 4 | SENTIMENT ANALYSIS | 6 |
| 5 | DATA INFO | 7 |
| 6 | PLATFORM / PROCEDURE | 8 |
| 7 | DATA PRE-PROCESSING PIPELINE | 9 |
| 8 | APPLYING SENTIMENT ANALYSIS | 12 |
| 9 | RESULT | 14 |
| 10 | DATA IN SQL SERVER | 15 |
| 11 | CHALLENGES / LIMITATIONS | 17 |
| 12 | SOURCE CODE | 18 |

**ABSTRACT**

To analyze the public emotions from Twitter Data Set towards the Demonetization event which was announced by the Prime Minister of India on November 8, 2016. The analysis will be done by using Sentiment analysis algorithms which help us to determine the overall opinion of the public on Demonetization.

**INTRODUCTION**

On November 8, 2016, Prime minister of India announced in an unscheduled national, that two high-value notes of Indian currency of 500 and 1000 is invalid with the immediate effect.

Result = 86% of existing currency became invalid

Aim = destroy counterfeit currency, reduce corruption, stop terrorism financing, etc.

Reserve Bank of India also made some rules and regulation for people that because of this event they can withdraw only some limited amount of money from their bank and ATMs. Which resulted in extremely long queues, chaos, and panic across the nation.

People reacted to the event on social media including twitter with their own opinion and views. So, for this project we will take twitter dataset – collection of tweets related to this event and perform sentiment analysis on it and get to know whether this event was positive, negative or neutral overall.

**STEPS OF APPLICATION**

* Data Collection
* Collection of our dataset will be done from the internet source.
* Data Preprocessing
* We will clean that dataset which will help us to apply our sentiment analysis algorithm on it.
* Sentiment Analysis
* We will calculate the polarity of our preprocessed data and check the sentiment result in the next step.
* Sentiment Results
* We will visualize the result of our sentiment analysis using the graphs and charts with some python libraries.
* Store Data and results to RDBMS
* At the end we will store our data and result to RDBMS and show filtered results one by one with applying SQL queries on it.

**SENTIMENT ANALYSIS**

What is Sentiment Analysis?

* Sentiment analysis is the process of analyzing digital text to determine if the emotional tone of the message is positive, negative, or neutral.
* It can be applied on the whole document of text, whole paragraph of text or it can also be applied on the text sentence.
* It will give us the result in: Positive, Negative or Neutral.

How many types of Sentiment Analysis are in General?

* 2 types
* Lexicon based.
* Machine Learning based.

What is Lexicon based Sentiment Analysis?

* It has some predefined or we can say word and weight dictionary, where it has some scores that help to calculate a sentence's polarity.
* Also known as “Rule-based Sentiment Analysis”

What is Machine Learning based Sentiment Analysis?

* A machine learning approach trains a classifier in a labelled dataset and predicts sentiments using the model it creates.

**DATA SET INFORMATION**

* Our Dataset has been taken from the internet source.
* Link to Dataset source: <https://www.kaggle.com/datasets/shan4224/demonetization-in-india>
* Dataset is the collection of all the tweets of the event of India’s demonetization 2016.
* It has more than 13 properties of each tweet.
  + Text (Tweets)
  + favorited
  + favoriteCount
  + replyToSN
  + created
  + truncated
  + replyToSID
  + id
  + replyToUID
  + statusSource
  + screenName
  + retweetCount
  + isRetweet
  + retweeted
* The total number of tweets is around 15000.
* Our Dataset file format is CSV.

**PLATFORM SETTING / SYSTEM CONFIGURATION:**

* I have used Anaconda Navigator to generate Jupyter Notebook file.
* I have downloaded that file on my local disk and edited that file and written it with VS Code.
* The language I have used is Python3.
* I have used SQL Server and SSMS Client for my RDBMS purposes.
* Libraries I have used with this project are:
  + **Re:**
    - The re module provides a set of powerful regular expression facilities, which allows you to quickly check whether a given string matches a given pattern (using the match function) or contains such a pattern (using the search function).
    - I have used re package to preprocess data with regular expression.
  + **Pandas:**
    - Pandas is a software library written for the Python programming language for data manipulation and analysis.
    - I have used Pandas to extract data on which we can apply our sentiment analysis.
  + **Textblob:**
    - Textblob can be used for complex analysis and working with textual data. When a sentence is passed into Textblob it gives two outputs, which are polarity and subjectivity.
    - Polarity is the output that lies between [-1,1], where -1 refers to negative sentiment and +1 refers to positive sentiment.
  + **Matplotlib:**
    - Matplotlib is a comprehensive library for creating static, animated, and interactive visualizations in Python.
    - I have used this library to make creative result with graphs and charts.
  + **Pyodbc:**
    - pyodbc is a Python open-source module that simplifies access to ODBC databases such as Oracle, MySQL, PostgreSQL, SQL Server, etc.
    - I have used this package to connect with my SQL Server and store the data and result to SQL Server Database.

**DATA PRE-PROCESSING PIPELINE**

* I have removed unnecessary columns from the CSV file and just kept 3 columns.
  + tweetText
  + date on which tweet was created.
  + screenname which is username of the twitter account which posted the tweet.

**raw\_file: demonetization\_tweets\_raw.csv (downloaded from the internet) :**

A screenshot of a computer

Description automatically generated

**Demonetization\_tweets.csv (file after I removed all the unnecessary columns):**

A screenshot of a computer

Description automatically generated

* Then I write down the below code for further pre-processing on the tweetText.

A screen shot of a computer code

Description automatically generated with low confidence

Step 1:

* We are extracting tweetText from the entire row data and converting them to lowercase only.

Step 2:

* After lowering the text I am removing all the mentioned username and twitter account names (@) from the text.

Step 3:

* Then I am removing all the link text which are starting with www and http

Step 4:

* After that I am removing all the punctuations like fullstop, comma, and quotation, any symbols from the text.

Step 5:

* From that text we are taking only alphanumeric text.

Step 6:

* After this we have defined some stopwords which I am removing from the entire text and at the end we will get our processed cleaned text data on which we can apply our Sentiment analysis algorithm.

**APPLYING SENTIMENT ANALYSIS**

* I have applied my sentiment analysis on cleaned text data.
* I have used textblob library’s polarity checking thing to check polarity score of my text.
* If the score is above 0, then our text is POSITIVE.
* If the score is below 0, then our text is NEGATIVE.
* If the score is exact 0, then our text is kind NEUTRAL.

**CODE:**

A picture containing text, screenshot, font

Description automatically generated

**RESULT:**

**A screen shot of text

Description automatically generated with low confidence**

**A screenshot of a computer screen

Description automatically generated with low confidence**

**VISUALIZING THE RESULT:**

* After that I have generated a bar graph and a pie chart to see our result in a better way.

**BAR GRAPH:**

**A picture containing text, screenshot, rectangle, number

Description automatically generated**

**PIE CHART:**

**A picture containing text, screenshot, font, diagram

Description automatically generated**

**DATA IN SQL SERVER**

* At the end I have stored the data and results in RDBMS SQL Server and see the filtered results in the SQL Server using SSMS client by performing some SQL queries.

A screenshot of a computer code

Description automatically generated with low confidence

* First query will retrieve all the tweets:

A screenshot of a computer

Description automatically generated with medium confidence

* Second query will retrieve all the tweets which are Positive:

A screenshot of a computer

Description automatically generated with medium confidence

* Third query will retrieve all the tweets which are Negative:

A screenshot of a computer

Description automatically generated with low confidence

* Fourth query will retrieve all the tweets which are Neutral:

A screenshot of a computer

Description automatically generated with medium confidence

**CHALLENGES / PROBLEMS / LIMITATION**

1. **Retrieving Data using Twitter API**
   * Twitter essential developer account limitation for searching previous 7 days of tweets.
   * Only Student and Enterprise accounts have a facility which can allow us to search beyond 7 days of search.
   * And since this event happened in 2016, I could not get much data from twitter in past 7 days, So I had to take data from internet, Kaggle site.
2. **Kaggle Dataset was not labeled.** 
   * I could not use Naïve Bayes method to complete my sentiment analysis. In order to apply Machine Learning based Sentiment analysis, we required to have labeled dataset.
3. **Lexicon-based sentiment analysis**
   * It doesn’t help in identifying tweets with sarcasm, grammar mistake, misspelling, etc.

**SOURCE CODE**

**Python Code:**

*import* re

*from* textblob *import* TextBlob

*import* matplotlib.pyplot *as* plt; plt.rcdefaults()

*import* numpy *as* np

*import* matplotlib.pyplot *as* plt

*import* pandas *as* pd

*import* pyodbc

totalPositiveTweets = 0

totalNegativeTweets = 0

totalNeutralTweets = 0

totalTweets=0

filepath="C:/Users/divij/Downloads/Final Project/demonetization-tweets.csv"

df = pd.read\_csv(filepath, *encoding*= 'unicode\_escape')

rowTweetsList = []

cleanTweetsList = []

result = []

cleanTweetColumn = []

stopwords = ["rt","for", "on", "an", "a", "of", "and", "in", "the", "to", "from"]

*for* i,row *in* df.iterrows():

    tweet=row[0].strip()

    tweetLowerCase = tweet.lower()

    tweetNoMentions = re.sub("@[A-Za-z0-9\_]+","", tweetLowerCase)

    tweetNoHttpLinks = re.sub(r"http\S+", "", tweetNoMentions)

    tweetNoWWWLinks = re.sub(r"www.\S+", "", tweetNoHttpLinks)

    tweetNoPunctuations = re.sub('[()!?]', ' ', tweetNoWWWLinks)

    tweetNoPunctuations = re.sub('\[.\*?\]',' ', tweetNoPunctuations)

    tweetWithOnlyAlphaNumeric = re.sub("[^a-z0-9]"," ", tweetNoPunctuations)

    tweetTokenization = tweetWithOnlyAlphaNumeric.split()

    tweetNoStopWords = [w *for* w *in* tweetTokenization *if* not w in stopwords]

    finalCleanTweet = " ".join(word *for* word *in* tweetNoStopWords)

    rowTweetsList.append(tweet)

    cleanTweetsList.append(finalCleanTweet)

    tweetAnalysis=TextBlob(finalCleanTweet)

    totalTweets = totalTweets+1

*if*(tweetAnalysis.sentiment.polarity>0):

        totalPositiveTweets=totalPositiveTweets+1

        result.append("Positive")

        cleanTweetColumn.append(finalCleanTweet)

*elif*(tweetAnalysis.sentiment.polarity<0):

        totalNegativeTweets= totalNegativeTweets+ 1

        result.append("Negative")

        cleanTweetColumn.append(finalCleanTweet)

*elif*(tweetAnalysis.sentiment.polarity==0):

        totalNeutralTweets = totalNeutralTweets + 1

        result.append("Neutral")

        cleanTweetColumn.append(finalCleanTweet)

df["cleanTweet"] = cleanTweetColumn

df["result"] = result

reader = df

filepath1="C:/Users/divij/Downloads/Final Project/demonetization-tweets\_new.csv"

df.to\_csv(filepath1)

objects = ('totalPositiveTweets','totalNegativeTweets','totalNeutralTweets')

y\_pos = np.arange(len(objects))

performance = [totalPositiveTweets,totalNegativeTweets,totalNeutralTweets]

plt.bar(y\_pos, performance, *align*='center', *alpha*=0.5)

plt.xticks(y\_pos, objects)

plt.ylabel('Total Number of Tweets')

plt.title('Twitter Sentiment Analysis - India\'s Demonetization Event (Bar Graph) \n')

plt.show()

plt.pie(performance,  *labels*=objects,*autopct*='%1.1f%%', *shadow*=False, *startangle*=90)

plt.axis('equal')

plt.title('Twitter Sentiment Analysis - India\'s Demonetization Event (Pie Chart) \n')

plt.show()

driver = 'SQL Server'

server = 'DIVIJJOBANPUTRA'

database = 'CIS593\_finalProject'

connection = pyodbc.connect(f'DRIVER={driver};'

                            f'SERVER={server};'

                            f'DATABASE={database};'

                            f'Trusted\_Connection=yes;')

filepath1="C:/Users/divij/Downloads/Final Project/demonetization-tweets\_new.csv"

cursor = connection.cursor()

drop\_table\_query = """

DROP TABLE IF EXISTS Twitter\_Sentiment\_Analysis

"""

create\_table\_query = """

CREATE TABLE Twitter\_Sentiment\_Analysis(

    created VARCHAR(MAX) NULL,

    userName VARCHAR(MAX) NULL,

    tweet VARCHAR(MAX) NULL,

    analysis VARCHAR(MAX) NULL,

)

"""

cursor.execute(drop\_table\_query)

cursor.execute(create\_table\_query)

df = pd.read\_csv(filepath1)

*for* index, row *in* df.iterrows():

    insert\_query = f"""

    INSERT INTO Twitter\_Sentiment\_Analysis (created, userName, tweet, analysis)

    VALUES (?,?,?,?)

    """

    params = (str(row['created']), str(row['screenName']), str(row['cleanTweet']), str(row['result']))

    cursor.execute(insert\_query, params)

*# Close the connection*

cursor.commit()

cursor.close()

connection.close()

**SQL QUERIES:**

Select \* from Twitter\_Sentiment\_Analysis;

Select \* from Twitter\_Sentiment\_Analysis

where analysis = 'Positive';

Select \* from Twitter\_Sentiment\_Analysis

where analysis = 'Negative';

Select \* from Twitter\_Sentiment\_Analysis

where analysis = 'Neutral';