**Sentiment Analysis on Demonetization of Twitter Tweets using PIG**

Lovely Yeswanth Panchumarthi

# Table of Contents

[Certificate 1](#_Toc150883100)

[Acknowledgements 2](#_Toc150883101)

[Table of Contents 3](#_Toc150883102)

[Abstract 4](#_Toc150883103)

[Abbreviations 5](#_Toc150883104)

[List of Tables 6](#_Toc150883105)

[List of Commands 7](#_Toc150883106)

[List of Figures 8](#_Toc150883107)

[1. Introduction 9](#_Toc150883108)

[1.1 Sentiment Analysis on Demonetization 10](#_Toc150883109)

[1.2 PIG: Working and Environment. 11](#_Toc150883110)

[2. Methodology 12](#_Toc150883111)

[2.1 Dataset 12](#_Toc150883112)

[2.2 Preprocessing 14](#_Toc150883113)

[2.2.1 Bag of Words 16](#_Toc150883114)

[2.2.2 Top 10 most frequent words and their counts from the tweets 17](#_Toc150883115)

[2.3 Procedure 19](#_Toc150883116)

[2.3.1 Sentiment Analysis 19](#_Toc150883117)

[3. Discussion 22](#_Toc150883118)

[4. Results 23](#_Toc150883119)

[5. Concluding Remarks 24](#_Toc150883120)

[6. Future Work 25](#_Toc150883121)

[7. References 26](#_Toc150883122)

# Abstract

This study aims to analyze the sentiment of tweets related to the topic of demonetization. The data consists of tweets stored in a CSV file, each with an associated ID and text. The sentiment analysis is performed by tokenizing the text of each tweet into individual words and matching these words with a dictionary of pre-assigned sentiment ratings. The sentiment rating for each tweet is calculated as the average sentiment rating of its constituent words. Tweets are then classified as positive or negative based on their average sentiment rating. The classified tweets are stored separately for further analysis. This sentiment analysis provides valuable insights into public opinion on the topic of demonetization. The results can be used for policy-making, public relations strategies, and understanding the public’s response to such significant economic measures.

# Abbreviations

SQL Structured Query Language

CSV Comma Seperated by Values

HDFS Hadoop Distributed File System

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# List of Tables

Table 1. Twitter Tweets Dataset………………………………………………………13

Table 2. AFINN txt file………………………….……………………………………...13

# List of Commands

Command 1. raw\_data = load '/demion\_analysis/demonetization-tweets.csv' using

PigStorage(','); …………………...……….………...………………………...6

Command 2. get\_details = foreach raw\_data generate $0 as id,$1 as text;……...…...…..7

Command 3. tokens = foreach get\_details generate id, text,

FLATTEN(TOKENIZE(text)) as words;…………………………...………7

Command 4. dictionary = load '/demion\_analysis/AFINN.txt' using PigStorage('\t')

as (word:chararray,rating:int);……………………………………………..8

Command 5. word\_ratings = join tokens by words left outer, dictionary by word using

'replicated';……………..9

Command 6. ratings = foreach word\_ratings generate tokens::id as id,tokens::text as

text, dictionary::rating as rating;…………………….………...…………..10

Command 7. group\_words = group ratings by (id,text);………………………………..11

Command 8. avg\_ratings = foreach group\_words generate group,AVG(ratings.rating)

as tweet\_rating;……………………………………………………………...12

Command 9. positive\_tweets = filter avg\_ratings by tweet\_rating > 0;

negative\_tweets = filter avg\_ratings by tweet\_rating < 0;……………..13

Command 10. store positive\_tweets INTO '/demion\_analysis/positive\_tweets' using

PigStorage(',');

store negative\_tweets INTO '/demion\_analysis/negative\_tweets' using

PigStorage(',');……………………………………………………………..14

# List of Figures

Figure 1. Number of Tweets Over Time ………………………...….…………………...14

Figure 2. Top 10 Users with the Most Tweets…………………...……………………….14

Figure 3. Distribution of Retweet Counts..……………………...……………………….15

Figure 4. Proportion of Retweets………………….………………...…….……………….15

Figure 5. Bag of Words Query..…….……………………………...…………...………….16

Figure 6. Bag\_Of\_Words…...……….……………………………...…………...………….17

Figure 7. Word Count Query……...……….……………………………...……………….18

Figure 8. Output Query.………………..…….……………………………...……………..18

Figure 9. Top\_10\_words Output……..………………………...….………………………18

Figure 10. Sandbox Login.……….………………….…………...…………...…………....19

Figure 11. Loading Dataset...................................................................................................19

Figure 12. Tokenization Query ………………..……….…………………….……………20

Figure 13. Filtering Query...……….……………………..………...……………………....21

Figure 14. Query Output ...……….…………………………...…...………………………21

Figure 15. positive\_tweets...……….……………………………....……………………....23

Figure 16. negative\_tweets...……….……………………………...………………………23

# Introduction

**Problem Statement:**

To examine the sentiments of citizens in our country during the period of "Demonetization," a sudden and drastic change that significantly impacted the people of India.

**Domain:**

On the 8th of November, the Prime Minister of India declared that the majority of the nation’s currency, specifically 86%, would become invalid in a span of 50 days. This involved the removal of all 500- and 1000-rupee notes, which were the most widely used denominations in the country, from circulation. In their place, a new 2000-rupee note was introduced. This action was presented as an effort to combat corruption and bring the largely untaxed informal economy under scrutiny.

**Motivation:**

The government's implementation of demonetization has significantly impacted the lives of individuals. Presently, social media platforms serve as the primary means for expressing emotions related to daily life issues and events. The viewpoints and emotions of the public play a crucial role in understanding the comprehensive impact of a sudden change in the country on individuals. This encourages us to explore the mental resilience of individuals in response to such circumstances. Analyzing and identifying the positive, neutral, and negative sentiments of citizens towards a major decision made by a Prime Minister has become exceptionally important.

**Solution:**

Numerous social media platforms offer a space for individuals to express their mental perspectives and ideas, with Twitter standing out as a prominent network. On Twitter, individual posts are referred to as tweets, and the platform sees millions of these daily from users worldwide. This vast pool of tweets provides a valuable resource for analyzing consumer opinions on specific situations. In recent years, these tweets have emerged as a crucial source of information essential for the success of brands, businesses, political careers, and for gauging public sentiment on various national decisions and situations.

In our project, we have employed Sentiment Analysis on tweets during the period of demonetization. Our approach involves extracting positive, neutral, and negative sentiments from these tweets using parts of speech analysis.

## [Sentiment Analysis on Demonetization](#_2s8eyo1)

In the ever-expanding landscape of big data analytics, the quest to distill actionable insights from vast datasets has led to the evolution of sophisticated tools and frameworks. Among these, Apache Pig stands out as a robust and user-friendly scripting platform built on top of the Hadoop ecosystem. Renowned for its simplicity and scalability, Pig enables the seamless processing of large datasets, making it an invaluable asset for projects dealing with extensive information. This project embarks on a compelling exploration, leveraging the capabilities of Pig to conduct sentiment analysis on the subject of demonetization.

Sentiment analysis, a powerful natural language processing technique, provides a nuanced understanding of public sentiments expressed across diverse digital platforms during the demonetization period. By categorizing sentiments into positive, negative, or neutral, we aim to uncover the multifaceted dimensions of opinions and emotions articulated through social media, news articles, blogs, and forums. In this pursuit, Apache Pig emerges as a facilitator, streamlining the process of data extraction, transformation, and analysis, ensuring efficiency and scalability in handling the voluminous textual data associated with this significant economic event.

As we delve into the intricacies of sentiment analysis on demonetization, the project not only seeks to unravel the sentiments surrounding this economic policy but also serves as a testament to the prowess of Apache Pig in the realm of big data analytics. By combining advanced analytics with the capabilities of Pig, we aim to contribute not only to the understanding of the societal impact of economic policies but also to the ongoing discourse on the role of innovative tools in unraveling intricate patterns within large datasets.

Demonetization, as witnessed in events such as India's bold move in 2016, involves the discontinuation of specific currency notes as legal tender, with far-reaching implications for the economy and society. This project aims to delve into the sentiments surrounding demonetization, unraveling the opinions, emotions, and reactions of the public through the lens of social media, news articles, blogs, and forums. By harnessing the capabilities of Pig, we aim to streamline the process of data processing and analysis, making it efficient and scalable for handling large volumes of textual data.

As we navigate the complexities of sentiment analysis on demonetization, the utilization of Pig facilitates seamless data processing, enabling us to distill meaningful patterns and trends from the vast textual data available. This project not only contributes to the broader understanding of the societal impact of economic policies but also showcases the efficacy of Apache Pig in handling sentiment analysis tasks, opening avenues for further exploration in the realm of big data analytics.

## PIG: Working and Environment.

Apache Pig is a high-level platform for creating MapReduce programs used with Hadoop. It is designed to process large data sets and is known for its simplicity and ease of use. Pig scripts use a language called Pig Latin, which is specifically designed for expressing data transformations in a more straightforward and flexible manner compared to traditional MapReduce programs. Apache Pig is designed to handle any kind of data—be it structured or unstructured—making it a versatile tool in the world of big data. It can manage data from various sources including databases, local files, and Hadoop’s Distributed File System (HDFS). Pig Latin, the scripting language used in Pig, is procedural and allows developers to develop their own functions for reading, processing, and writing data. This makes Pig highly extensible and adaptable to various data processing tasks.

Pig has several uses in data processing tasks. It can handle various data types, including structured and unstructured data, and it supports complex data operations like filters, joins, ordering, and more. It’s particularly useful for performing ETL (Extract, Transform, Load) tasks, data analysis, and iterative data processing.

Pig’s ability to handle parallel execution of tasks makes it a powerful tool for handling large data sets in distributed systems. Its simplicity and flexibility make it a popular choice for data scientists and engineers working with big data. It allows users to focus more on analyzing bulk data sets and spend less time writing Map-Reduce programs. This makes Apache Pig an essential tool in the Hadoop ecosystem.

One of the key strengths of Pig is its ability to handle parallel processing, which is crucial when dealing with large datasets. It automatically optimizes tasks and splits them into sub-tasks that can be run in parallel across a Hadoop cluster. This feature, combined with the simplicity of Pig Latin, allows developers to write complex data transformations without having to think about the underlying architecture or optimization.

# Methodology

## 

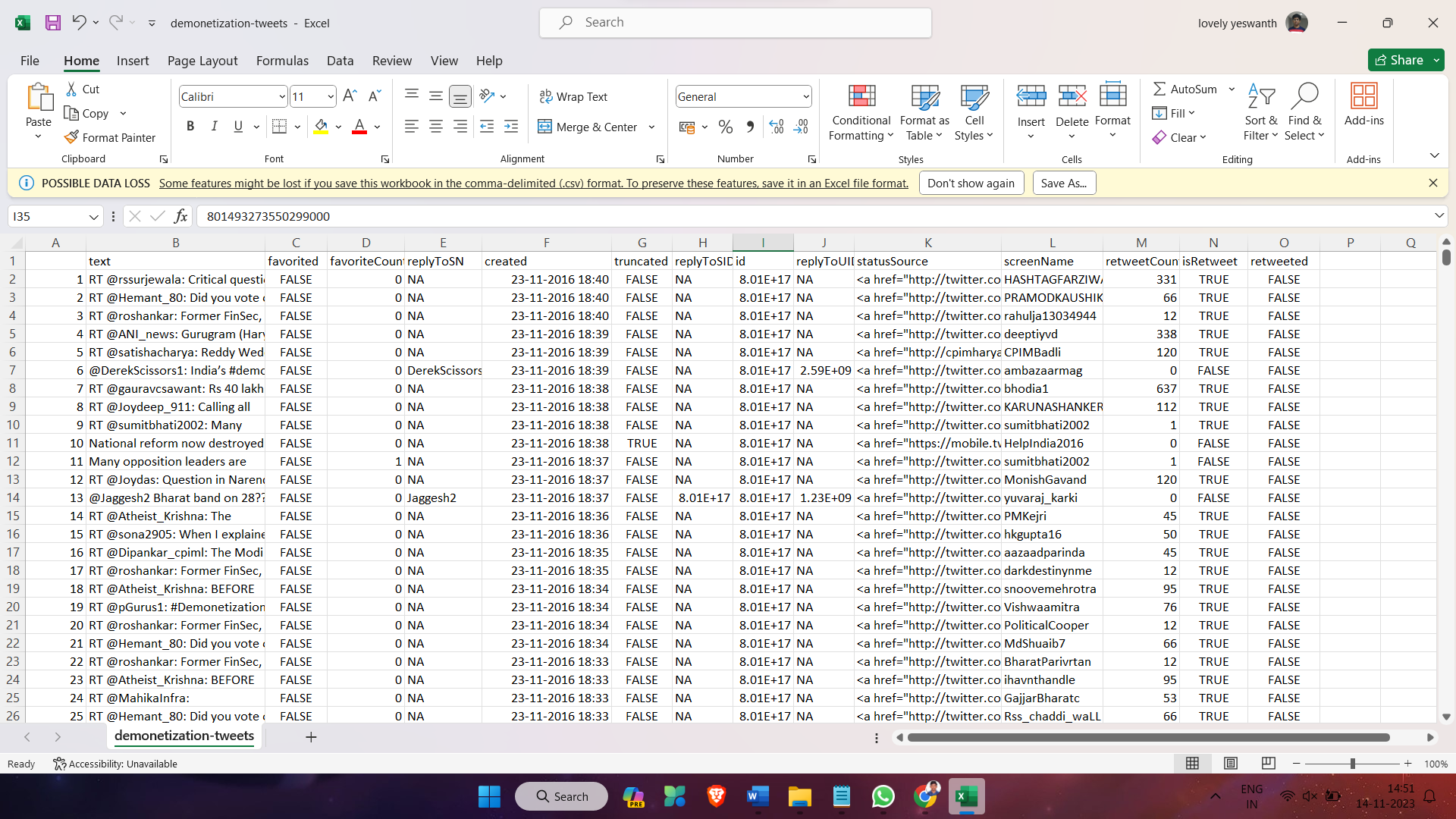
## Dataset

The ‘demonetization-tweets.csv’ dataset is a collection of tweets related to the topic of demonetization. Each row in the dataset represents a single tweet and contains the following fields:

1. **text**: This is the actual text of the tweet. It contains the message that the user posted on Twitter.
2. **favorited**: This is a boolean field indicating whether the tweet has been favorited by the user.
3. **favoriteCount**: This field shows the number of favorites the tweet has received.
4. **replyToSN**: This field contains the screen name of the user to whom the tweet is replying.
5. **created**: This field contains the date and time when the tweet was created.
6. **truncated**: This is a boolean field indicating whether the tweet text has been truncated.
7. **replyToSID**: This field contains the status ID of the tweet to which the current tweet is replying.
8. **id**: This field contains the unique ID of the tweet.
9. **replyToUID**: This field contains the user ID of the tweet to which the current tweet is replying.
10. **statusSource**: This field contains the source of the tweet, indicating from which device or application the tweet was posted.
11. **screenName**: This field contains the screen name of the user who posted the tweet.
12. **retweetCount**: This field shows the number of times the tweet has been retweeted.
13. **isRetweet**: This is a boolean field indicating whether the tweet is a retweet.
14. **retweeted**: This is a boolean field indicating whether the tweet has been retweeted by the user.

This dataset provides a rich source of information for analyzing public sentiment towards demonetization. By processing and analyzing these tweets, one can gain insights into the public’s reaction to demonetization, the spread of information, and the influence of different users in the Twitter community. This dataset is particularly useful for sentiment analysis, topic modeling, social network analysis, and other data mining tasks.

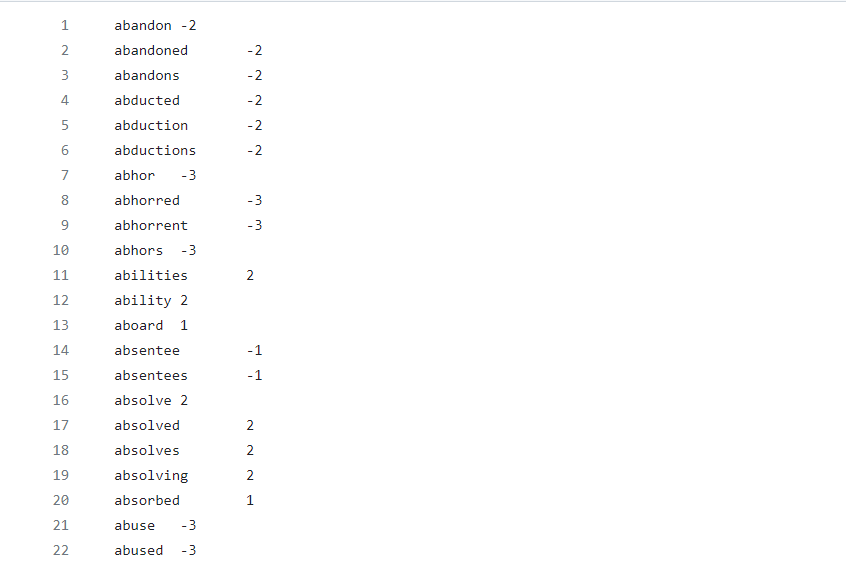
Link to the dataset online: <https://www.kaggle.com/shan4224/demonetization-in-india>



**Table 1. Twitter Tweets Dataset**

We will load the dictionary into pig by using the below statement:

The AFINN lexicon is a list of English terms manually rated for valence with an integer between -5 (negative) and +5 (positive) by Finn Årup Nielsen between 2009 and 2011.



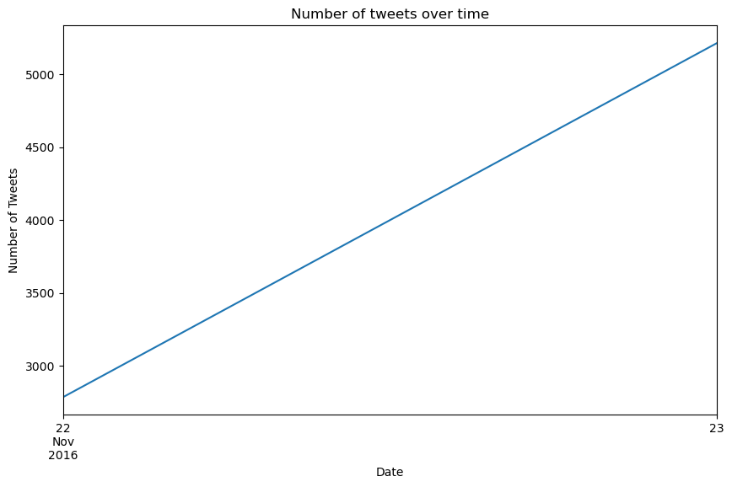
**Table 2. AFINN txt file**

The original lexicon contains some multi-word phrases,

Link for AFINN: <http://corpustext.com/reference/sentiment_afinn.html>

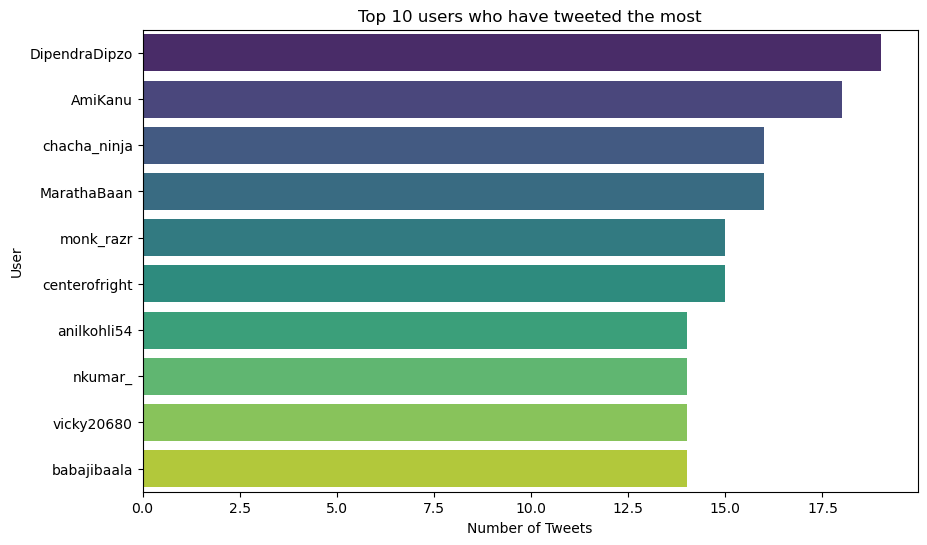
## Preprocessing

**1. Number of Tweets Over Time:** The time series plot illustrates the temporal distribution of tweets related to demonetization. The x-axis represents dates, with the data resampled by day, and the y-axis shows the corresponding number of tweets on each day. This visualization provides insights into patterns, trends, and potential spikes in tweet activity over the observed period.



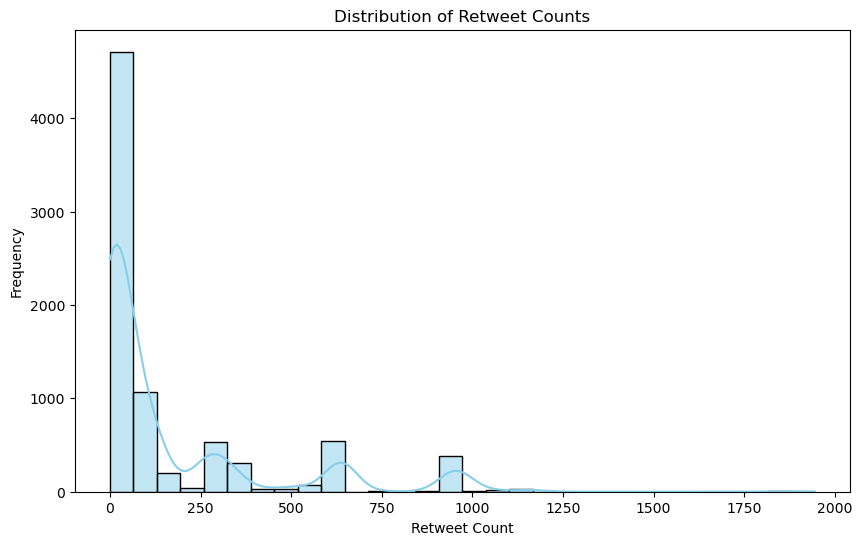
**Figure 1. Number of Tweets Over Time**

**2. Top 10 Users with the Most Tweets:** The horizontal bar plot highlights the top 10 users who have contributed the most tweets related to demonetization. Each bar represents a user, and its length corresponds to the number of tweets they have posted. This visualization helps identify influential users or those actively participating in the discourse on demonetization.



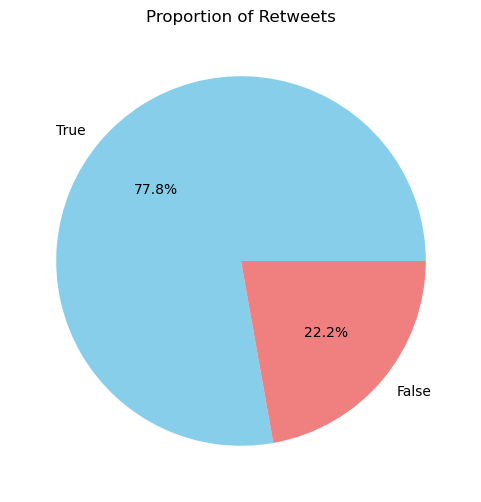
**Figure 2. Top 10 Users with the Most Tweets**

**3. Distribution of Retweet Counts:** The histogram and kernel density estimate plot the distribution of retweet counts across all tweets. The x-axis represents the number of retweets, and the y-axis indicates the frequency of tweets falling within each retweet count range. This visualization provides insights into the spread and concentration of retweet activity, revealing whether most tweets receive a low or high number of retweets.



**Figure 3. Distribution of Retweet Counts**

**4. Proportion of Retweets:** The pie chart displays the proportion of retweets compared to original tweets in the dataset. Each slice represents either a retweet or an original tweet, and the percentage labels indicate the relative contribution of each type. This visualization offers a quick overview of the distribution between retweets and original content, helping to understand the extent to which users engage in amplifying existing tweets versus creating new ones.



**Figure 4. Proportion of Retweets**

### Bag of Words

Pig script that could be used for topic modeling on the 'demonetization-tweets.csv' dataset. Pig itself doesn't support advanced machine learning algorithms like LDA. You would typically use a tool like Apache Mahout or Spark MLlib for topic modeling. However, you can use Pig to preprocess the data before feeding it into the topic modeling algorithm.

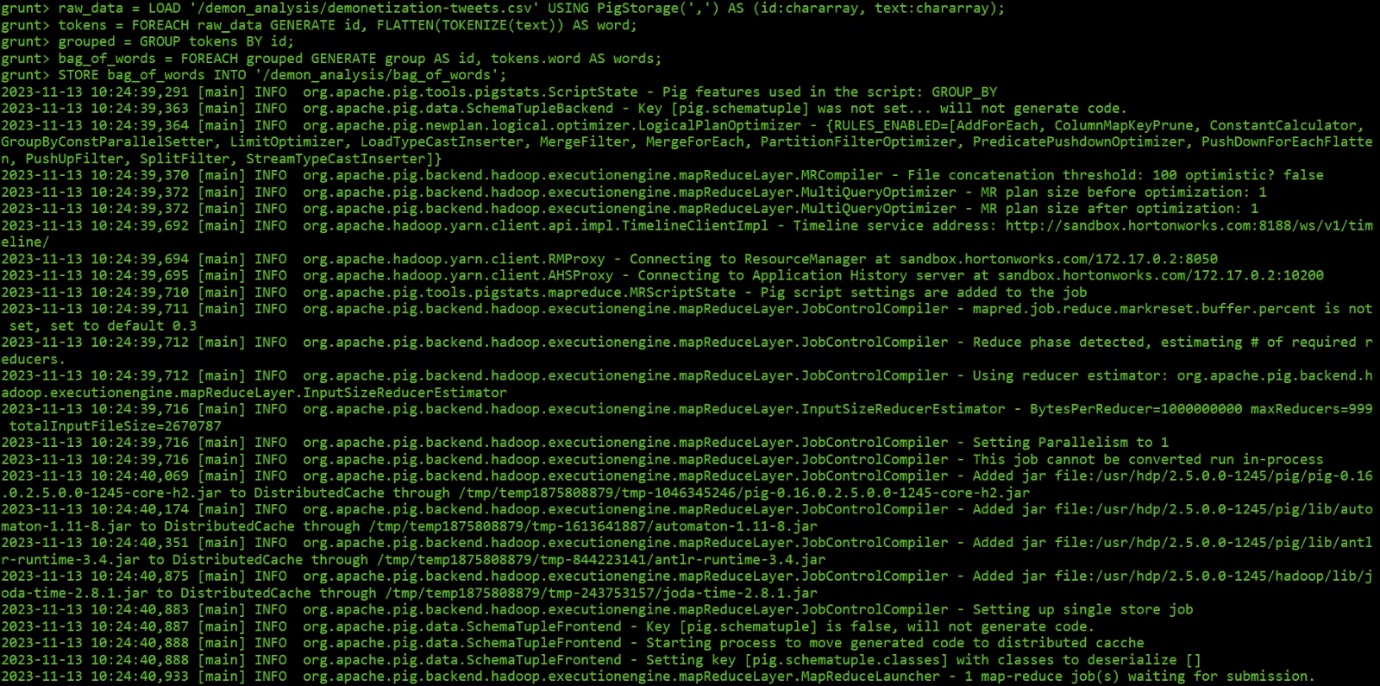
**Load the tweets:** raw\_data = LOAD '/demion\_analysis/demonetization-tweets.csv' USING PigStorage(',') AS (id:chararray, text:chararray);

**Tokenize the text of each tweet:** tokens = FOREACH raw\_data GENERATE id, FLATTEN(TOKENIZE(text)) AS word;

**Group the tokens by tweet id:** grouped = GROUP tokens BY id;

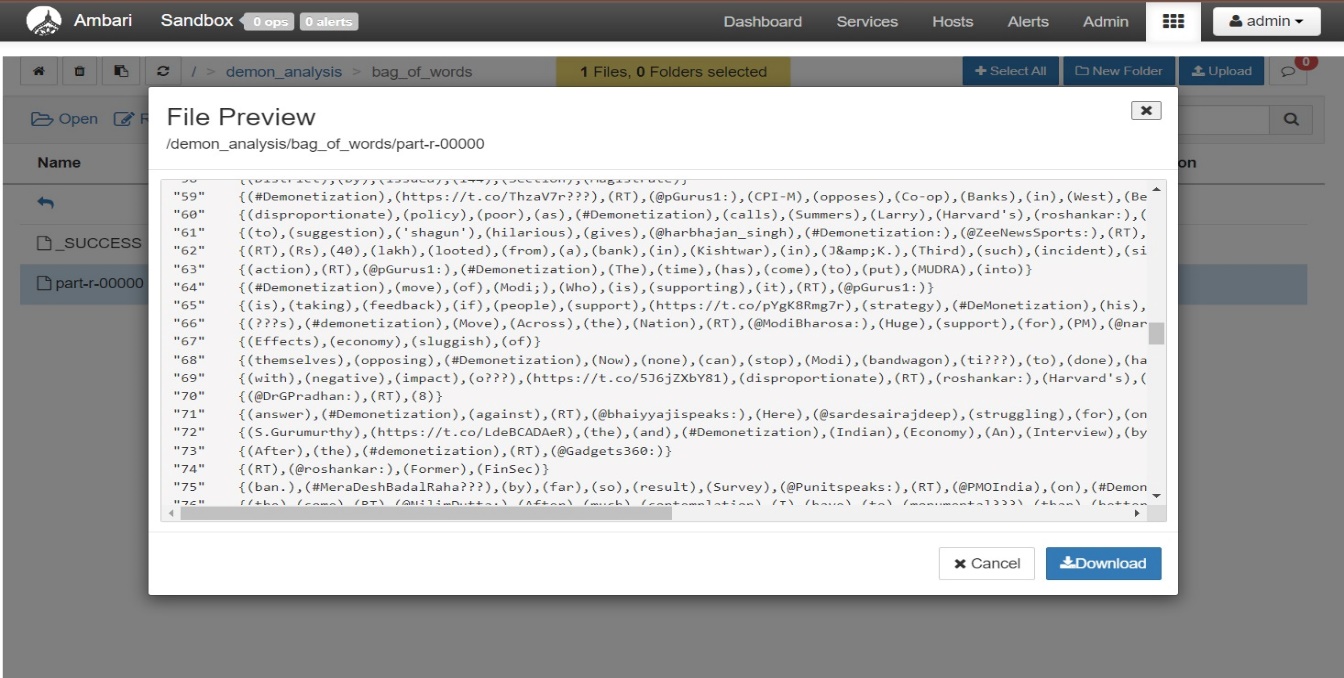
**For each tweet, create a bag of words:** bag\_of\_words = FOREACH grouped GENERATE group AS id, tokens.word AS words;

**Store the result:** STORE bag\_of\_words INTO '/demion\_analysis/bag\_of\_words';



**Figure 5. Bag of Words Query**

This script will create a 'bag of words' for each tweet, which can then be used as input for the LDA algorithm. The LDA algorithm can be applied using a tool like Apache Mahout or Spark MLlib. The output of the LDA algorithm will be a set of topics, each represented as a distribution over words, and each tweet will be represented as a distribution over these topics. This can provide insights into the main topics of discussion in the tweets related to demonetization.

**Figure 6. Bag\_Of\_Words**

### Top 10 most frequent words and their counts from the tweets

A Pig script to find the top 10 most frequent words and their counts from the tweets in the demonetization-tweets.csv file. Load the data from the file into a relation called tweets: tweets = LOAD ‘/Project/demonetization-tweets.csv’ USING PigStorage(‘,’) AS (id:long, text:chararray, favorited:boolean, favoriteCount:int, replyToSN:chararray, created:chararray, truncated:boolean, replyToSID:long, replyToUID:long, statusSource:chararray, screenName:chararray, retweetCount:int, isRetweet:boolean, retweeted:boolean);

**Extract the tweet text from the relation:** tweet\_text = FOREACH tweets GENERATE text;

**Split the tweet text into words:** words = FOREACH tweet\_text GENERATE FLATTEN(TOKENIZE(text)) AS word;

**Convert the words to lowercase:** words\_lower = FOREACH words GENERATE LOWER(word) AS word;

**Filter out the words that are not alphabetic:** words\_alpha = FILTER words\_lower BY word MATCHES ‘\\w+’;

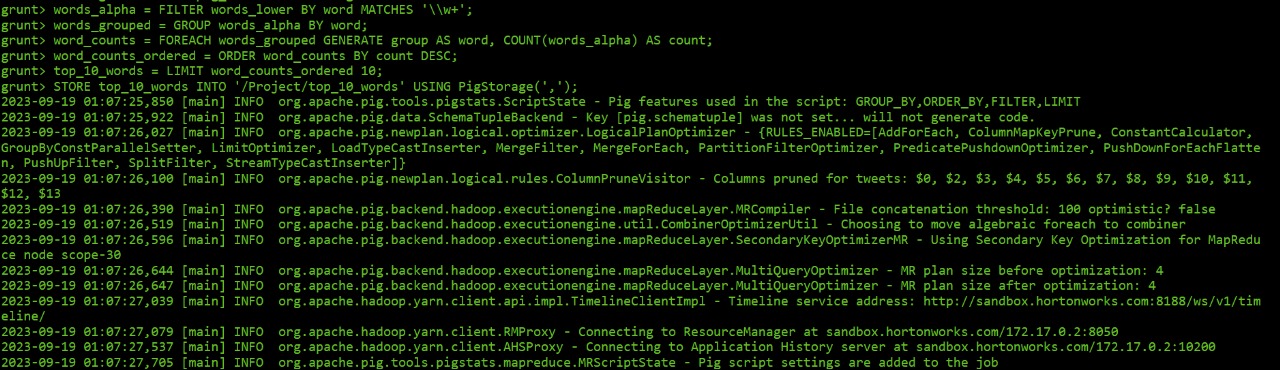
**Group the words by word:** words\_grouped = GROUP words\_alpha BY word;

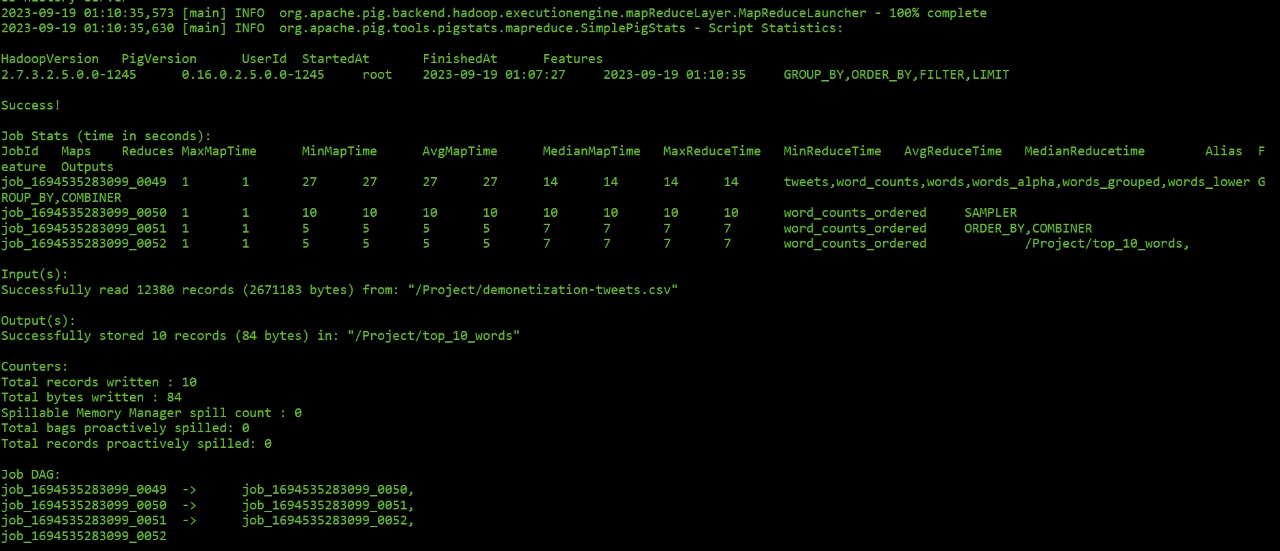
**Count the number of occurrences of each word:** word\_counts = FOREACH words\_grouped GENERATE group AS word, COUNT(words\_alpha) AS count;

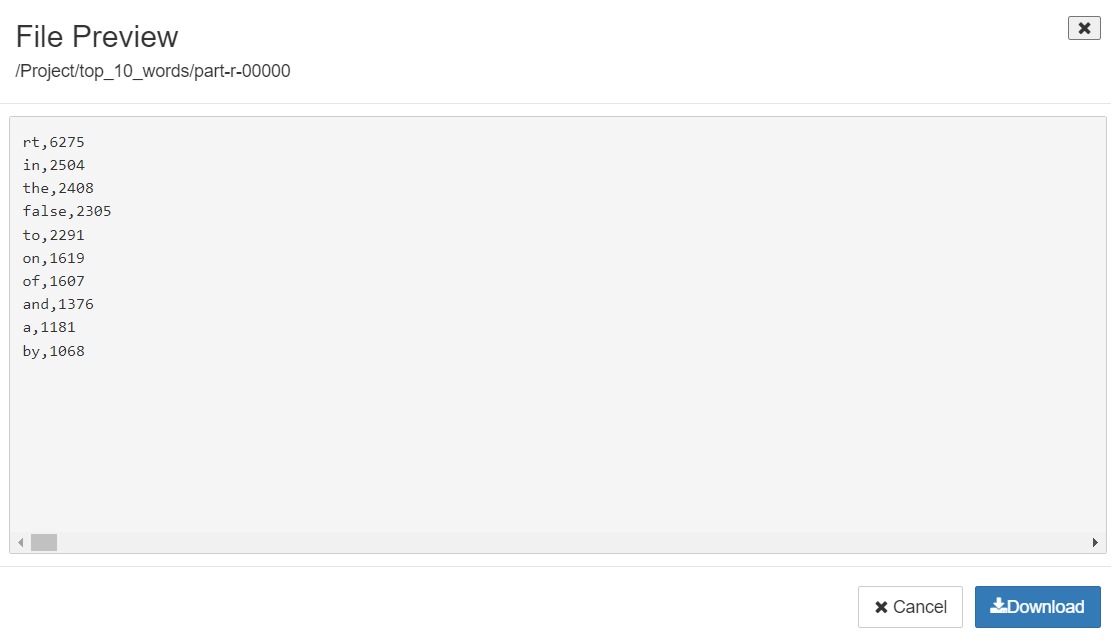
**Order the word counts by count in descending order:** word\_counts\_ordered = ORDER word\_counts BY count DESC;

**Limit the result to the top 10 words:** top\_10\_words = LIMIT word\_counts\_ordered 10;

**Store the result into a file called top\_10\_words.csv:** STORE top\_10\_words INTO ‘/demion\_analysis/top\_10\_words’ USING PigStorage(‘,’);

**Figure 7. Word Count Query**

**Figure 8. Output Query**

****

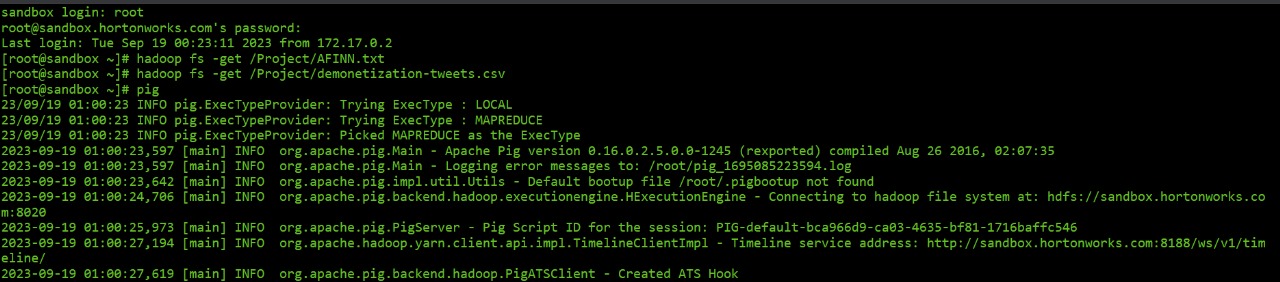
**Figure 9. Top\_10\_words Output**

## Procedure

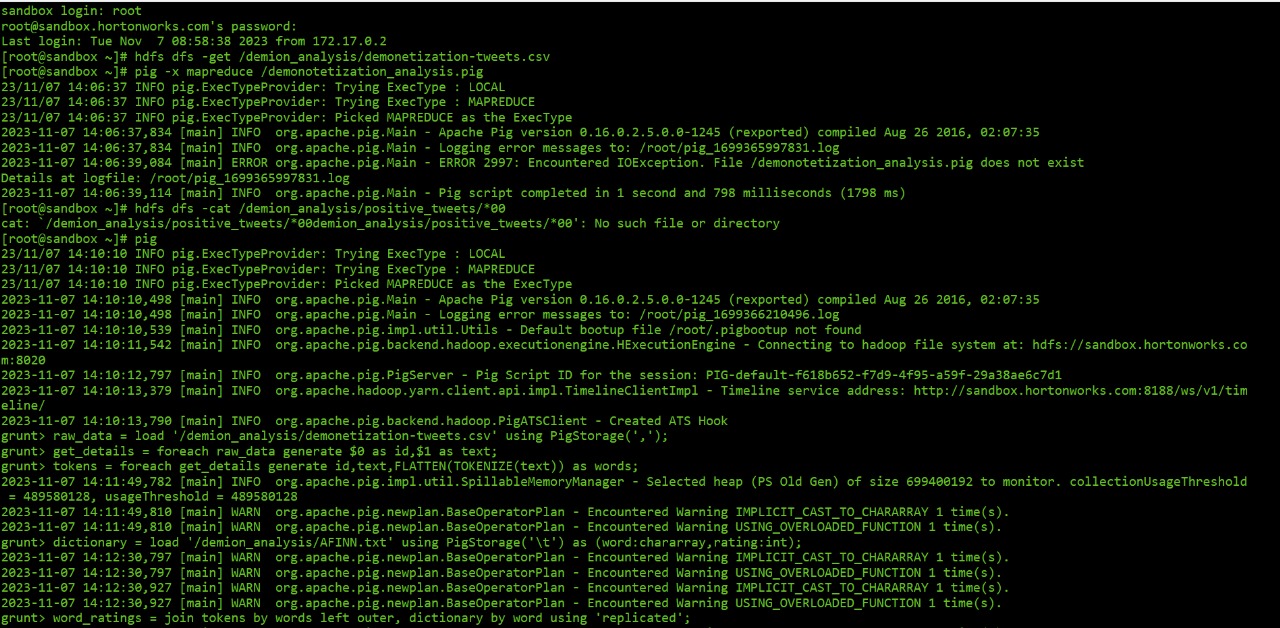
### Sentiment Analysis

**1. Data Ingestion and HDFS Setup:** Begin by setting up the Hadoop Distributed File System (HDFS) environment. Upload the provided CSV file **("demonetization-tweets.csv")** into the newly created folder using the ‘get’ command.

**2. Launching Pig:** Open the PIG shell by typing `**pig**` in the **command line** interface.



**Figure 10. Sandbox Login**

**3. Loading the Data**: The first line of the script loads the tweet data from a CSV file located at ‘/demion\_analysis/demonetization-tweets.csv’. The load function is used to load the data and PigStorage is used to specify the delimiter (comma in this case). The loaded data is stored in the raw\_data relation.

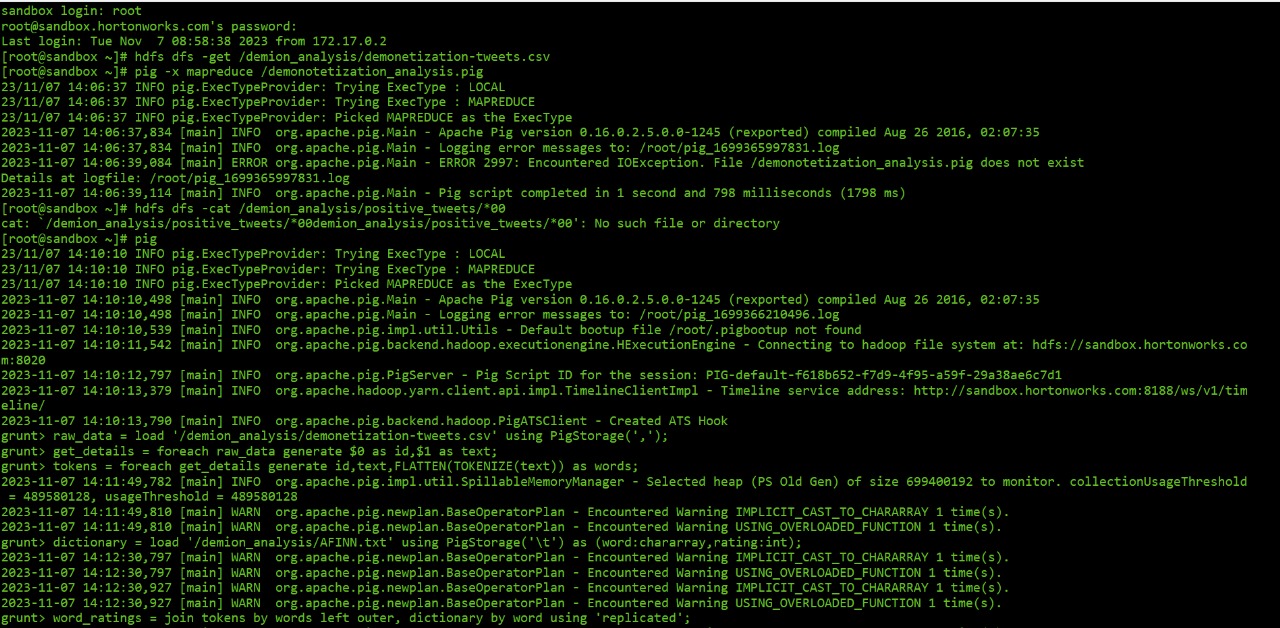
**Figure 11. Loading Dataset**

**4. Extracting Tweet Details**: The second line uses the foreach function to iterate over each record in raw\_data and generate a new relation get\_details that contains the ID and text of each tweet.

**5. Tokenizing the Text**: The third line tokenizes the text of each tweet into individual words using the TOKENIZE function. The FLATTEN function is used to ensure that each word gets its own record. The result is stored in the tokens relation.

**6. Loading the Dictionary**: The fourth line loads the sentiment dictionary from a text file located at ‘/demion\_analysis/AFINN.txt’. Each record in the dictionary consists of a word and its associated sentiment rating. The loaded dictionary is stored in the dictionary relation.

**7. Joining the Tokens and Dictionary**: The fifth line performs a join operation between the tokens and dictionary relations based on the word. The join function is used for this purpose. The result of the join operation is stored in the word\_ratings relation.



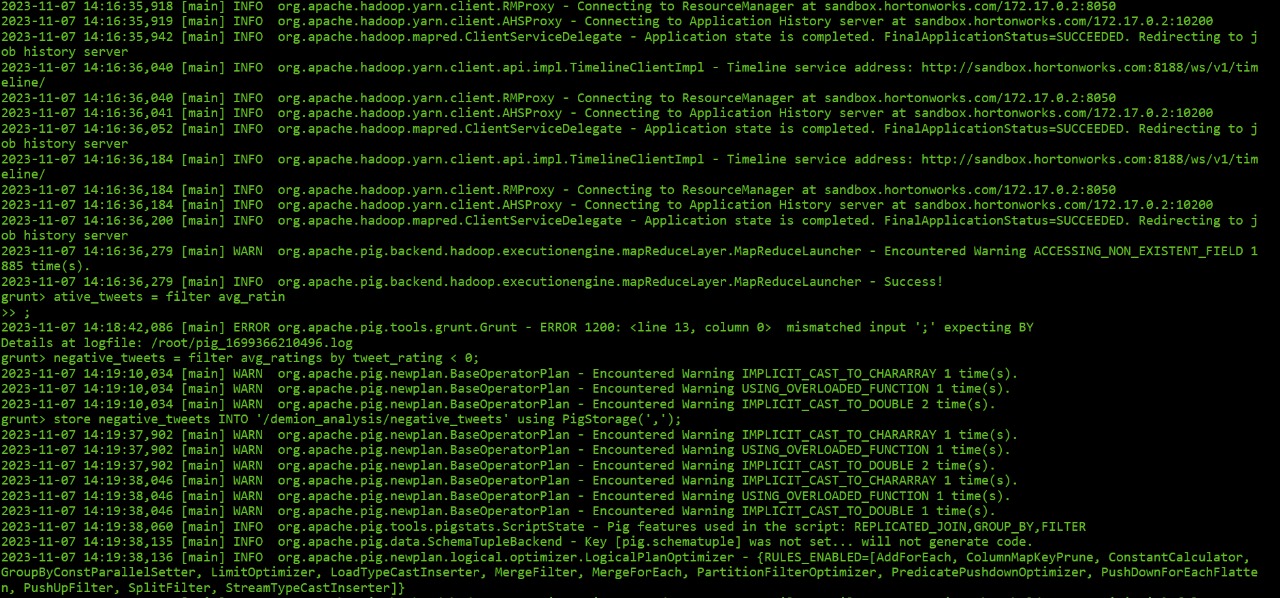
**Figure 12. Tokenization Query**

**8. Generating Ratings**: The sixth line generates a new relation rating that contains the ID and text of each tweet along with the sentiment rating of each word in the tweet.

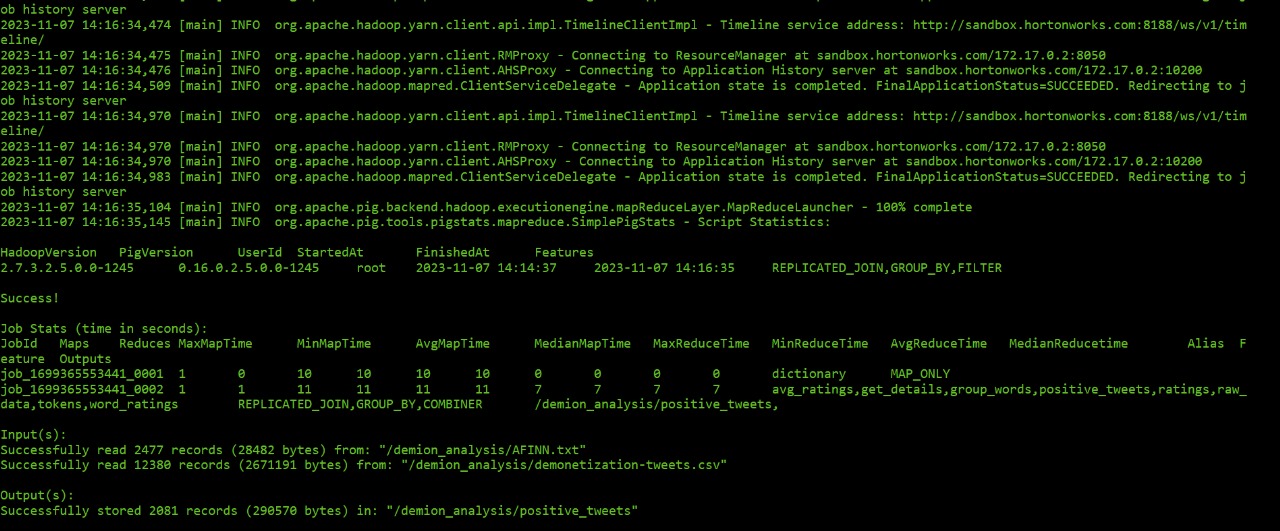
**9. Grouping the Words**: The seventh line groups the ratings relation by tweet ID and text. The group function is used for this purpose. The result is stored in the group\_words relation.

**10. Calculating Average Ratings**: The eighth line calculates the average sentiment rating for each tweet using the AVG function. The result is stored in the avg\_ratings relation.

**11. Filtering Positive and Negative Tweets**: The ninth and tenth lines filter the avg\_ratings relation to separate positive and negative tweets. Tweets with an average sentiment rating greater than 0 are considered positive, while those with an average sentiment rating less than 0 are considered negative. The filter function is used for this purpose.

**Figure 13. Filtering Query**

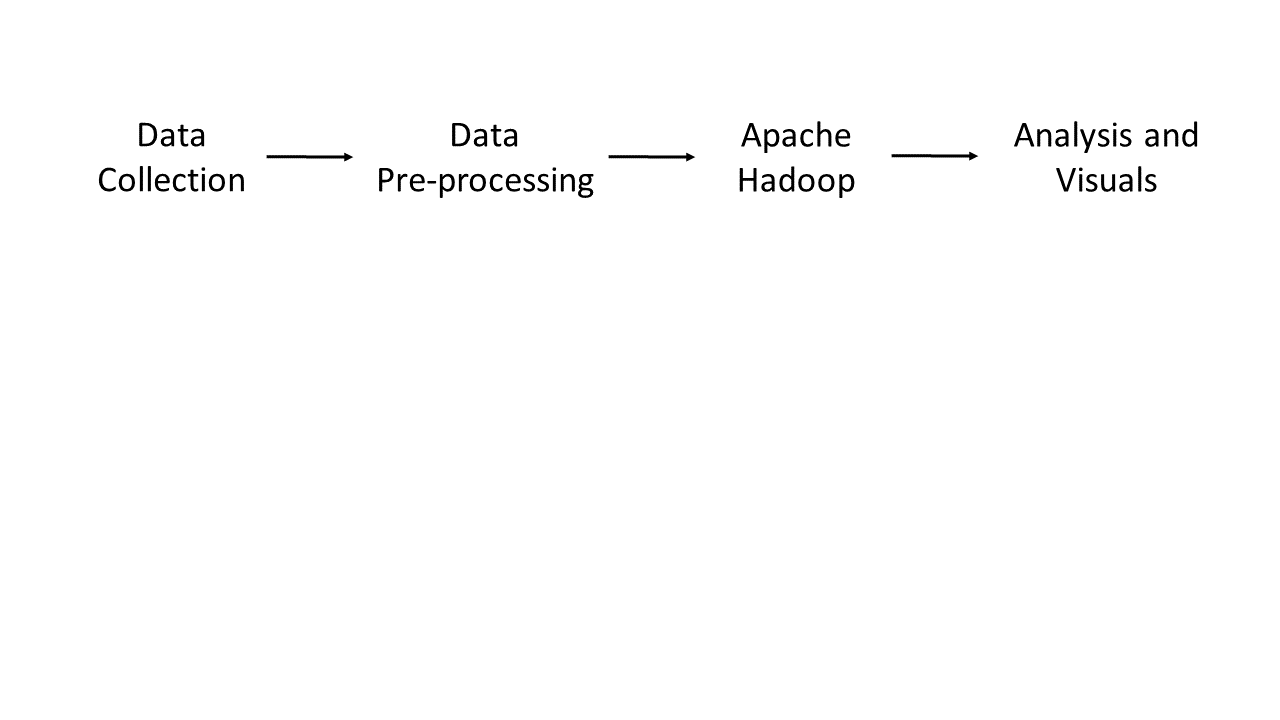
**12. Storing the Results**: The last two lines store the positive and negative tweets into separate CSV files located at ‘/demion\_analysis/positive\_tweets’ and ‘/demion\_analysis/negative\_tweets’ respectively. The store function is used to store the results.

**Figure 14. Query Output**

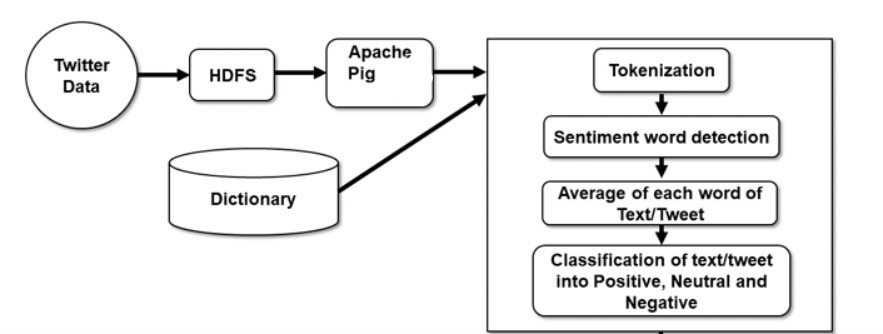
# Discussion

**Architecture and Design of the System**

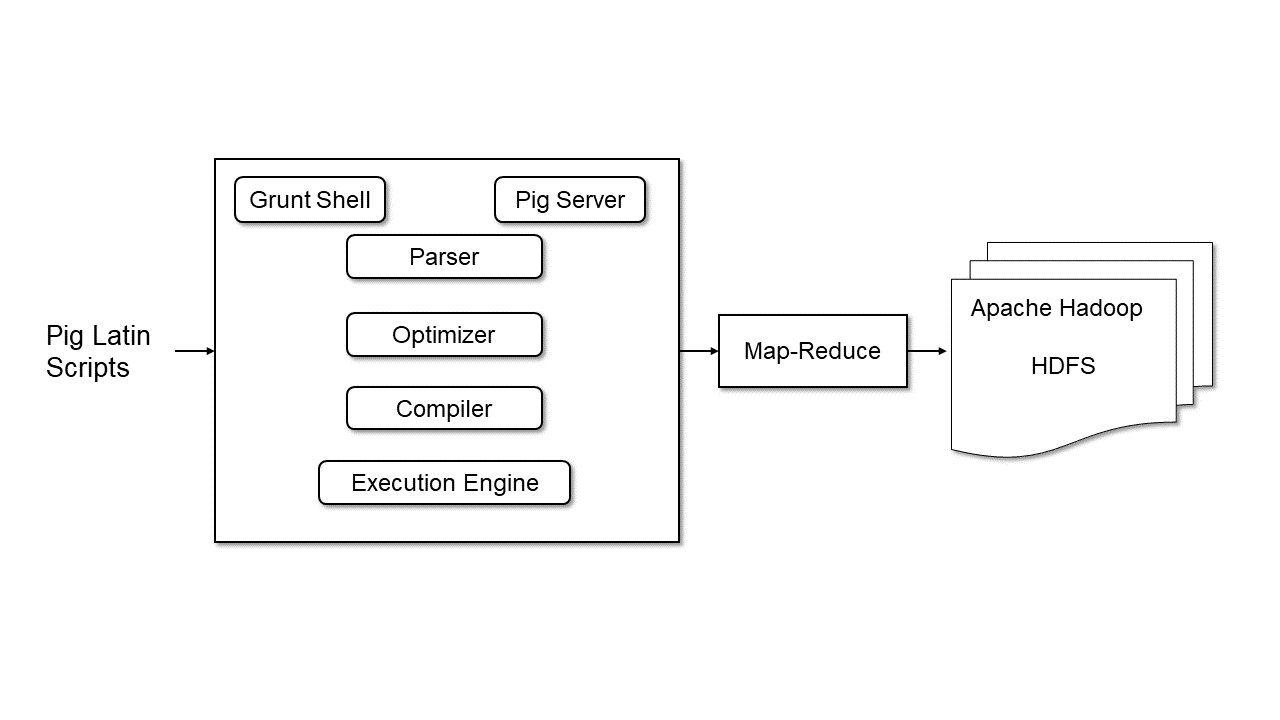
1. Process Flow:



1. Architecture Design:



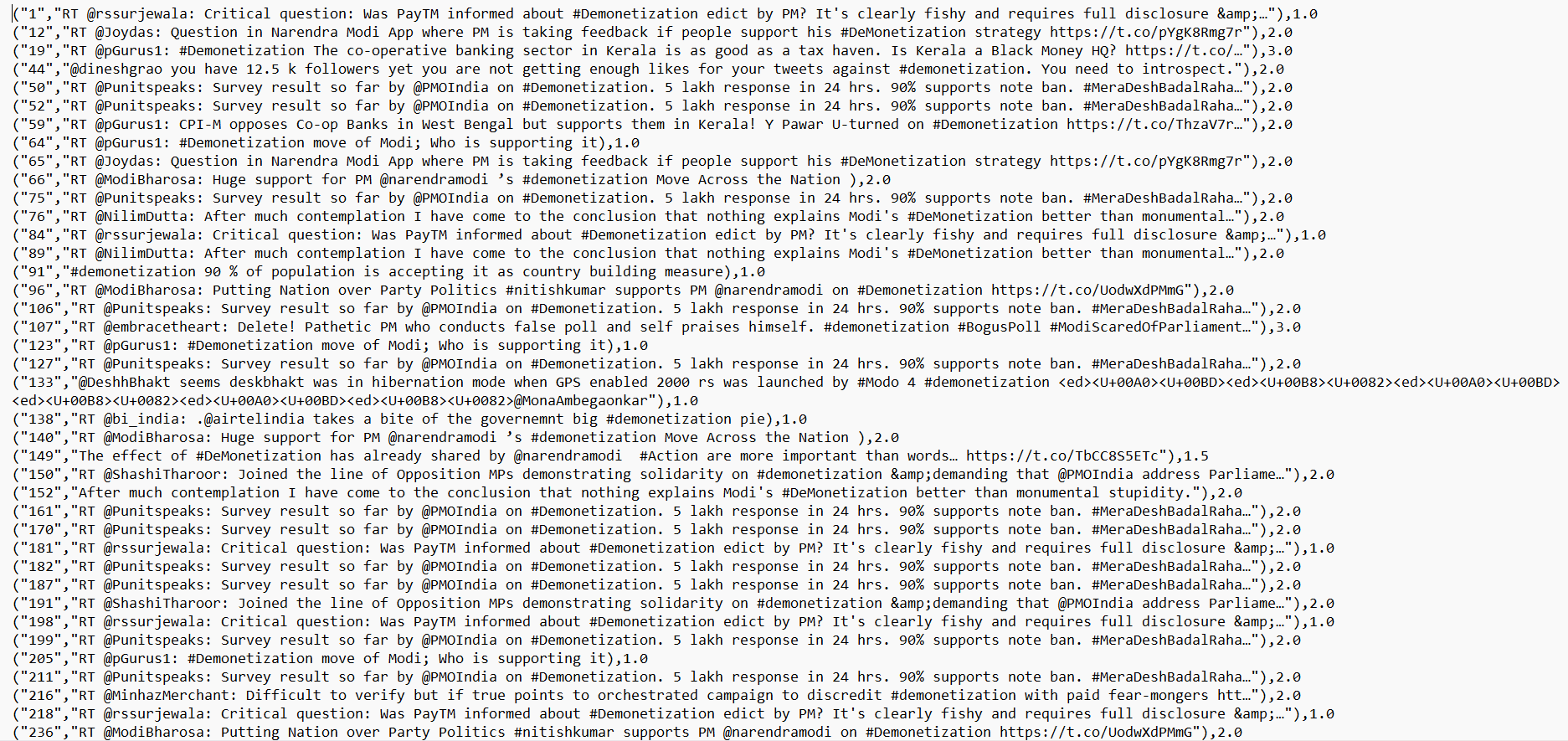
1. Pig framework:

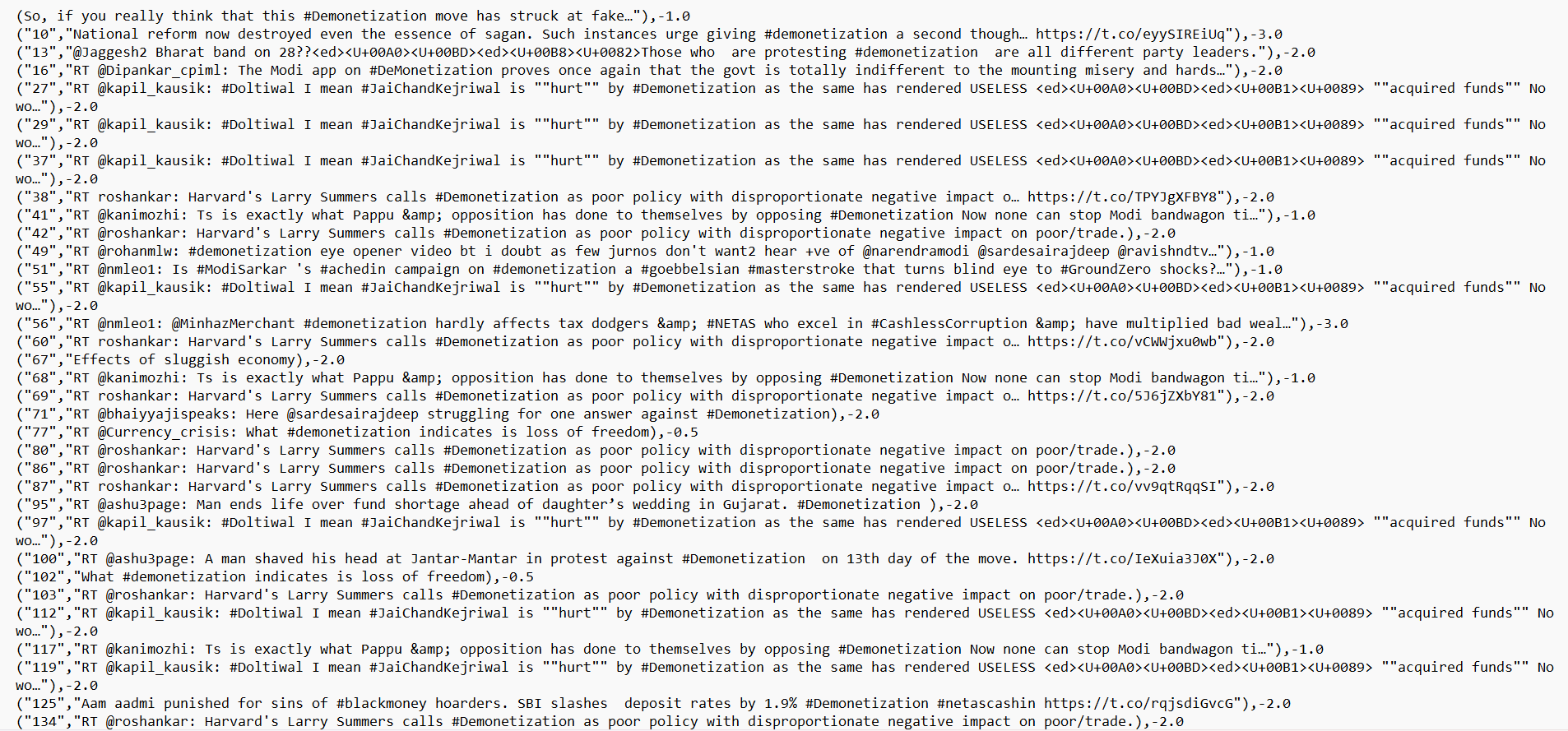


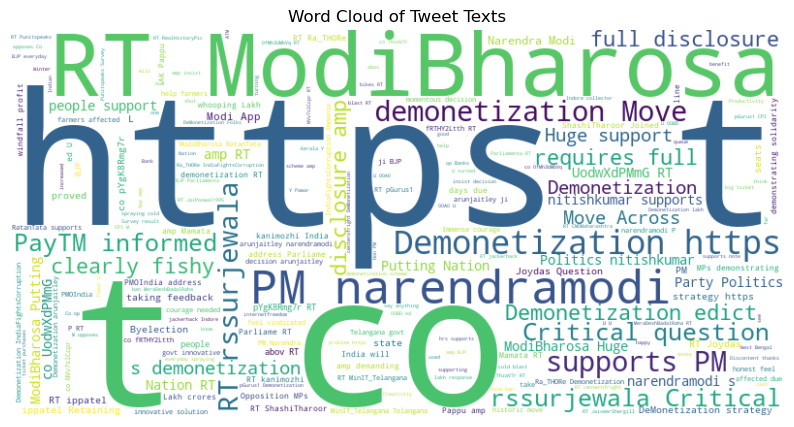
Characteristics:

1. Hive queries are implicitly converted to the mapper and reducer job. Also, extensible, scalable and fast which helps in analysis.
2. Unstructured or semi-structured data cannot be problem as Pig is used.
3. Hadoop itself is fault-tolerance.

# Results

**Figure 15. positive\_tweets**

**Figure 16. negative\_tweets**

**Figure 17. Word Cloud of Positive Tweets**

# Concluding Remarks

In conclusion, the project successfully utilized Apache Pig to perform sentiment analysis and topic modeling on a dataset of tweets related to demonetization. The Pig scripts developed for this project enabled efficient processing of large volumes of data, transforming raw tweets into a structured format suitable for analysis.

The sentiment analysis provided valuable insights into public opinion on demonetization, categorizing tweets as positive or negative based on their content. This could potentially be used to gauge public sentiment towards policy decisions, providing a valuable tool for decision-makers.

The topic modeling further enriched our understanding of the discussions around demonetization. By identifying the main topics of discussion, we were able to uncover the key themes and concerns raised by the public in their tweets.

Additionally, the project also involved finding the most frequent words in the tweets. This helped in identifying trending topics and commonly used words in the discussions around demonetization.

Overall, this project demonstrated the power and versatility of Apache Pig in handling big data tasks. It showcased how a high-level language like Pig Latin can simplify the process of writing complex MapReduce programs, making it easier to extract meaningful insights from large datasets. The findings from this project could potentially inform future research and decision-making processes related to similar topics.

The success of this project underscores the potential of big data technologies like Apache Pig in transforming raw, unstructured data into valuable insights. As we continue to generate and have access to increasing volumes of data, tools like Apache Pig will become increasingly important in helping us understand and make sense of this information.

# Future Work

For future work, there are several potential avenues to explore:

**Enhanced Sentiment Analysis**: The sentiment analysis could be enhanced by incorporating more sophisticated Natural Language Processing (NLP) techniques. For instance, the use of deep learning models such as Recurrent Neural Networks (RNNs) or Transformers could potentially improve the accuracy of sentiment classification.

**Temporal Analysis**: An interesting extension would be to perform a temporal analysis of the tweets. This would involve analyzing how the sentiment and topics of discussion evolved over time. This could provide insights into how public opinion changed in response to various events or announcements related to demonetization.

**User-Level Analysis**: Another potential area of exploration is user-level analysis. This would involve analyzing the activity and influence of individual users. For example, one could identify the most influential users in the discussion of demonetization, or analyze how the sentiment of a user’s tweets changes over time.

**Geographical Analysis**: If the geographical data of the tweets (such as the location from where the tweet was posted) is available, it would be interesting to perform a geographical analysis. This could reveal regional differences in sentiment and topics of discussion related to demonetization.

**Integration with Other Data Sources**: The analysis could be enriched by integrating the tweet data with other data sources. For example, economic indicators or news events could be correlated with the sentiment and topics of the tweets to gain a deeper understanding of the public’s reaction to demonetization.

**Improving Topic Modeling**: The topic modeling could be improved by using more advanced techniques or by fine-tuning the parameters of the LDA algorithm. Additionally, other topic modeling algorithms could be explored.

These are just a few ideas for future work. The possibilities for extending this project are vast, given the richness of the data and the wide array of techniques available for analyzing it. The insights gained from such extensions could provide even more valuable insights into public opinion on important topics like demonetization.

# References

1. 2016. Opinion Mining of Twitter Data using Hadoop and Apache Pig. International Journal of Computer Applications. Volume 146 – No.11.
2. 2019. A survey on sentiment analysis methods, applications, and challenges. Journal of Big Data. Volume 6, Article number: 107.
3. 2020. Sentiment Analysis with Apache Pig. GitHub Repository.
4. 2020. COVID-19 sentiment analysis via deep learning during the rise of novel COVID-19 cases in India. Journal of Big Data. Volume 7, Article number: 110.
5. 2017. Sentiment Analysis on Demonetization Tweets Pig. Medium Blog Post.
6. 2018. Weather Dataset Analysis Using Apache Pig. International Journal of Computer Applications. Volume 180 – No.21.
7. 2019. Analyzing Performance of Apache Pig and Apache Hive with Hadoop. International Journal of Computer Applications. Volume 179 – No.21.
8. 2018. Apache Pig for Big Data Analysis. Medium Blog Post.
9. 2016. Count word occurrences in each row using Pig. Stack Overflow Post.
10. 2017. Sentiment Analysis of twitter data using hadoop and pig. Stack Overflow Post.
11. 2017. Word Count Example in Pig. Riptutorial.
12. 2016. Pig Tutorial | Twitter Case Study | Apache Pig Script and Commands. SlideShare Presentation.