VISVESVARAYA TECHNOLOGICAL UNIVERSITY
BELAGAVI-590014

R Mini Project
Report On

Text Mining Using R Programming

A Project report submitted in partial fulfillment of the requirements for the VIII Semester degree of
Bachelor of Engineering in Computer Science and Engineering
of Visvesvaraya Technological University, Belagavi

Submitted by

SAMARTH SRIVASTAVA	**(1DT21CS131)**
SHIVAM ANAND	**(1DT21CS143)**
SHUBHAM TULSYAN	**(1DT21CS147)**
SIDDHARTH KUMAR	**(1DT21CS148)**

Under the Guidance of

Mrs. Shilpa M
Assistant Professor
Computer Science and Engineering
Dayananda Sagar Academy of Technology & Management

Department of Computer Science and Engineering
DAYANANDA SAGAR ACADEMY OF TECHNOLOGY AND MANAGEMENT
Udayapura, Kanakapura Road, Bangalore-560082

2022-2023

TABLE OF CONTENT

ABSTRACT

Text mining is a powerful technique in data analysis that involves extracting valuable information and insights from textual data. In this report, we explore the use of R programming for text mining, discussing its importance, methods, and practical applications. We provide an introduction to text mining, highlighting its relevance in today's data-driven world.

R packages like 'tm', ' SnowballC', 'NLP', and ' wordcloud2' are utilized for data preprocessing, feature extraction, and model building. Sentiment analysis tools help in understanding the emotional tone of the text, while topic modeling reveals the underlying themes and subjects. Named entity recognition identifies and categorizes important entities like names, locations, and organizations within the text.

Through this project, valuable insights are gained from the analyzed text data, which can be used for various applications such as market research, customer feedback analysis, and content recommendation systems. The power of R in text mining is demonstrated, showcasing its ability to handle and extract knowledge from vast amounts of unstructured textual information.

INTRODUCTION

In today's digital age, vast amounts of textual data are generated daily, ranging from social media posts and customer reviews to scientific articles and business documents. This proliferation of text data presents a unique opportunity to extract valuable information and insights from unstructured text. Text mining, also known as text analytics or natural language processing (NLP), is the process of converting unstructured text into structured data for analysis and interpretation.

R programming has emerged as a powerful tool for text mining due to its extensive libraries and packages tailored to NLP tasks. In this report, we will delve into the world of text mining in R, exploring its significance and practical applications.

Text Mining in R

R programming offers a comprehensive ecosystem for text mining, with several essential packages:

1. **tm (Text Mining)** : The tm package provides a framework for text preprocessing, transformation, and analysis. It includes functions for text cleaning, tokenization, and term-document matrix creation.
2. **SnowballC** : Snowball Stemmers Based on the C 'libstemmer' UTF-8 Library. An R interface to the C 'libstemmer' library that implements Porter's word stemming algorithm for collapsing words to a common root to aid comparison of vocabulary.
3. **NLP** : The NLP package provides various natural language processing functionalities, including part-of-speech tagging, stemming, and lemmatization.
4. **wordcloud2** : It is a visual representation of text data. The size of each word indicates its frequency or importance. In R, you can create word clouds using the wordcloud package.

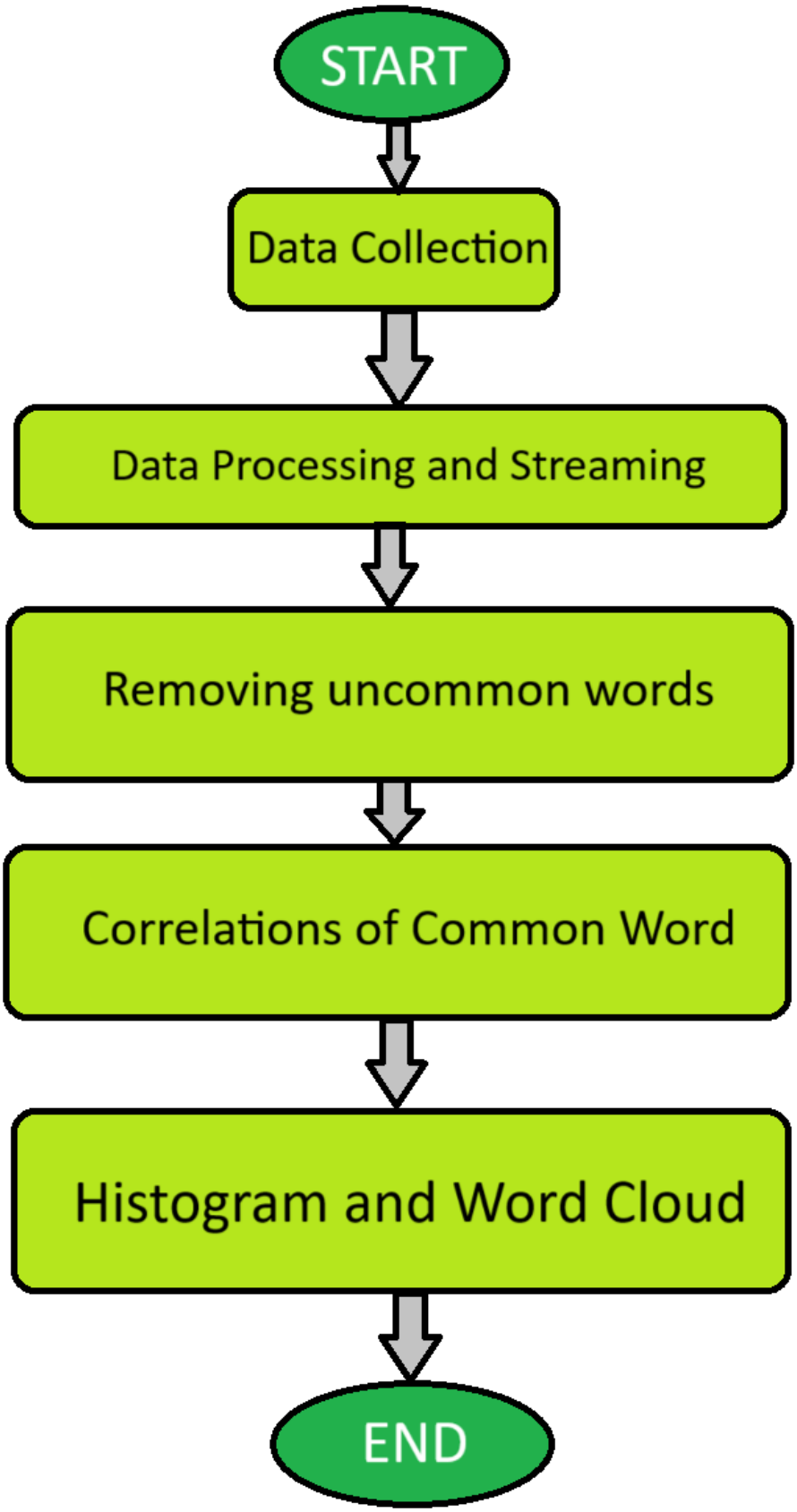
PROGRAM CODE

```
#Initial Code
{
  #Start
  {
  #install.packages('tm')
  library(tm)

  docs<- Corpus(DirSource('Base2/'))

  toSpace <- content_transformer(function(x, pattern) {return(gsub(pattern, "
",x))})

  docs <- tm_map(docs, toSpace, "-")
  docs <- tm_map(docs, toSpace, ":")
  docs <- tm_map(docs, toSpace, "'")
  docs <- tm_map(docs, toSpace, '"')
  docs <- tm_map(docs, toSpace, " -")

  docs<- tm_map(docs, removePunctuation)
  docs <- tm_map(docs,content_transformer(tolower))
  docs <- tm_map(docs, removeNumbers)
  docs <- tm_map(docs, removeWords, stopwords("english"))
  docs <- tm_map(docs, stripWhitespace)
}
  # Stemming
  {
  #install.packages('SnowballC')
  library(SnowballC)

  docs <- tm_map(docs, content_transformer(gsub), pattern = "activity", replacement
="active")
  docs <- tm_map(docs, content_transformer(gsub), pattern = "ting", replacement
="te")
  docs <- tm_map(docs, content_transformer(gsub), pattern = "ning", replacement
="n")
  docs <- tm_map(docs, content_transformer(gsub), pattern = "stories", replacement
="story")

  docs <- tm_map(docs,stemDocument)

  docs <- tm_map(docs, content_transformer(gsub), pattern = "challeng", replacement
="challenge")
  docs <- tm_map(docs, content_transformer(gsub), pattern = "creativ", replacement
="creative")
  docs <- tm_map(docs, content_transformer(gsub), pattern = "stori", replacement
="story")
  docs <- tm_map(docs, content_transformer(gsub), pattern = "easi", replacement
="easy")
```

```
  docs <- tm_map(docs, content_transformer(gsub), pattern = "forc", replacement
="force")
  docs <- tm_map(docs, content_transformer(gsub), pattern = "undertaken",
replacement ="undertake")
  docs <- tm_map(docs, content_transformer(gsub), pattern = "websit", replacement
="website")
  docs <- tm_map(docs, content_transformer(gsub), pattern = "comput", replacement
="computer")
  docs <- tm_map(docs, content_transformer(gsub), pattern = "electr", replacement
="electric")
  docs <- tm_map(docs, content_transformer(gsub), pattern = "messag", replacement
="message")
  docs <- tm_map(docs, content_transformer(gsub), pattern = "devic", replacement
="device")
  docs <- tm_map(docs, content_transformer(gsub), pattern = "amaz", replacement
="amaze")
  docs <- tm_map(docs, content_transformer(gsub), pattern = "outsid", replacement
="outside")
}
  #Removing rare words
  {
  dtm <- DocumentTermMatrix(docs)

  freq <- colSums(as.matrix(dtm))

  ord <- order(freq, decreasing=TRUE)

  dtmr <-DocumentTermMatrix(docs, control=list(wordlengths=c(4,20), bounds =
list(global = c(2,27))))

  freqr = colSums(as.matrix(dtmr))

  ordr <- order(freqr, decreasing = TRUE)

  print("Words with frequency greater then 7")
  findFreqTerms(dtmr,lowfreq=7)
}
}
#correlations of Common Word
{
  print(findAssocs(dtmr, "computer" , 0.7))

  print(findAssocs(dtmr,"data",0.7))

  print(findAssocs(dtmr,"device",0.7))
}
#Histogram
{
  library(ggplot2)
  dtma <-DocumentTermMatrix(docs, control=list(wordlengths=c(2,20), bounds =
list(global = c(5,27))))
```

```
  m <- as.matrix(dtma)
  m
  Frequency <- sort(colSums(as.matrix(dtma)), decreasing=TRUE)
  wf <- data.frame(Words=names(Frequency), freq=Frequency)

  p <- ggplot(subset(wf, Frequency>1), aes(Words, Frequency))
  p <- p + geom_bar(stat="identity")
  p <- p + theme(axis.text.x=element_text(angle=45, hjust=1))
  p
}
#Word Cloud
{
  library(wordcloud)
  set.seed(42)
  wordcloud (names(freqr), freqr,min.freq=6,colors=brewer.pal (6, "Dark2"))
}
```

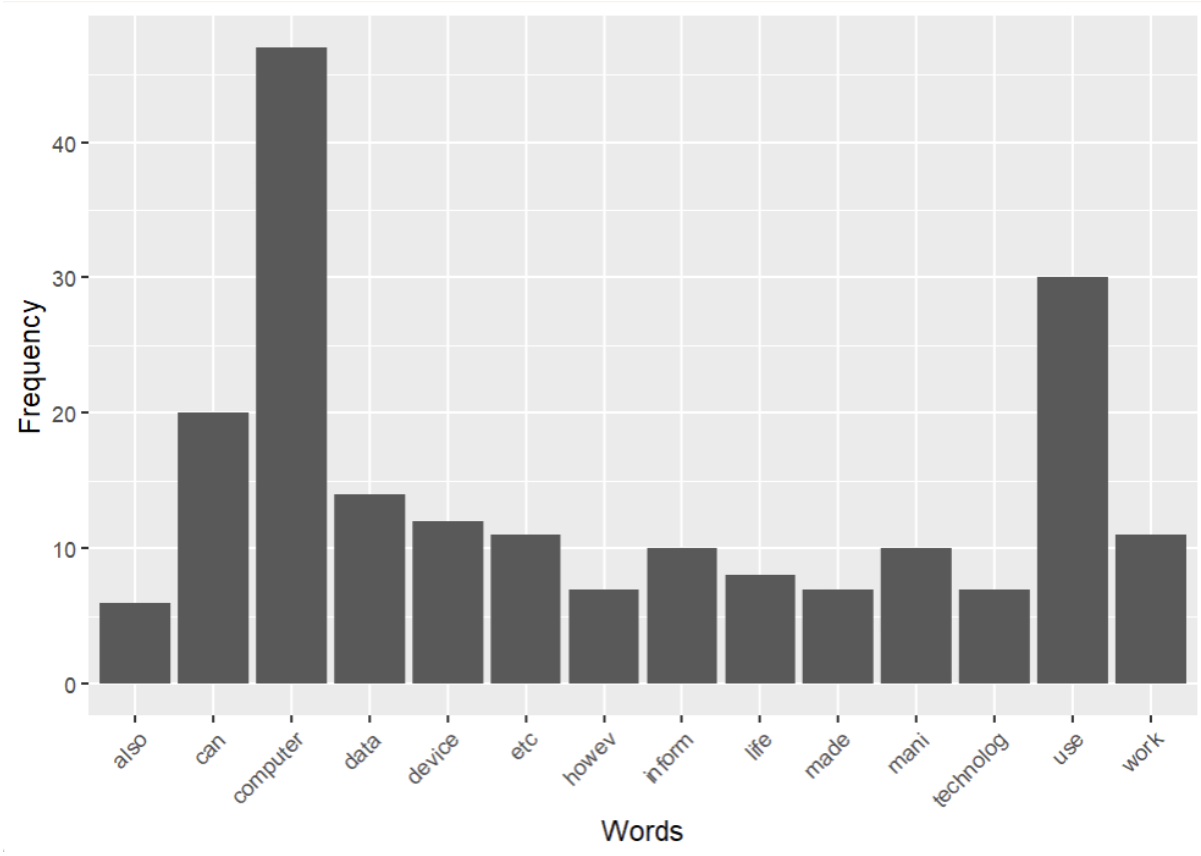
FLOW CHART


START
Data Collection
Data Processing and Streaming
Removing uncommon words
Correlations of Common Word
Histogram and Word Cloud
END

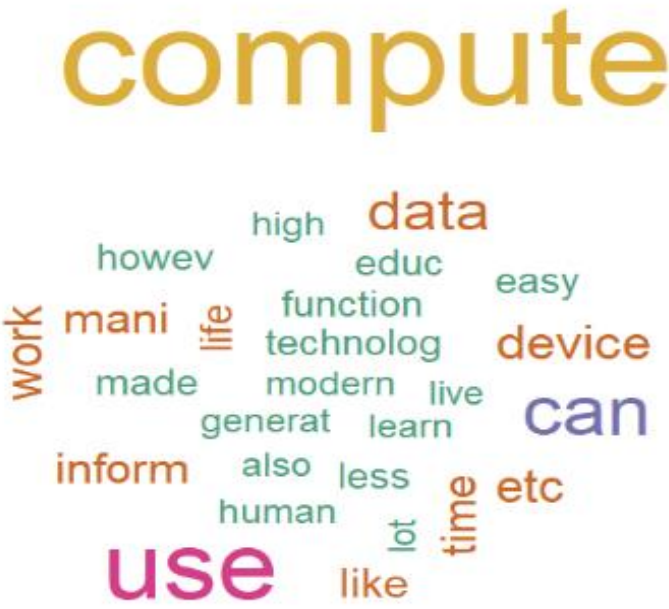
OBSERVED OUTPUTS

```
[1] "Words with frequency greater then 7"
 [1] "can"       "computer"  "data"      "device"    "educ"      "etc"       "function"  "like"
 [9] "technolog" "use"       "work"      "easy"      "howev"     "inform"    "less"      "life"
[17] "made"      "mani"      "time"


$computer
    day   world  advanc  babbag   charl     cpu   creat   first  mechan    mous outside softwar
   0.88    0.83    0.82    0.78    0.78    0.78    0.78    0.78    0.78    0.78    0.78    0.74
 invent
   0.72

$data
  device    input  printer   anytim  anywher   capabl    chang  message keyboard    simpl   output
    0.97     0.92     0.92     0.82     0.82     0.82     0.81     0.81     0.75     0.75     0.71
    bill
    0.71

$device
    data    input  printer  message    chang     bill   anytim  anywher   capabl   babbag    charl
    0.97     0.94     0.91     0.91     0.84     0.81     0.76     0.76     0.76     0.76     0.76
     cpu    creat    first   mechan     mous  outside keyboard   output
    0.76     0.76     0.76     0.76     0.76     0.76     0.74     0.73
```

OUTPUT SCREENSHOTS



Histogram of words with high frequency



Word cloud of word with high frequency

CONCLUSION

The conclusion of a text mining project using R should summarize the key findings, insights, and implications of your analysis. Here's a general structure for a conclusion in a text mining project using R:

Summary of the Project: This project finds all the words in given text files and determines the most occurring words and their probability of having in relation with all other words.

Limitations: We are using ‘SnowballC’ package for stemming which does not work properly as there are some errors with word ending with ‘es’.

To overcome this, we are manually modifying this package to make it suitable to our need.
We can also use Natural Language Tool Kit (NLTK) which is a python library for Stemming.

Future Scope : The future scope for text mining is characterized by the ongoing evolution of technology and its expanding applications. With advanced AI and NLP techniques, text mining can offer deeper insights into textual data, including sentiment analysis and context understanding.
Industries across healthcare, finance, marketing, and more will adopt text mining for data-driven decision-making. Integration with IoT and sensor data, along with educational initiatives, will further broaden its impact. In essence, the future of text mining holds great promise for enhancing information extraction and knowledge discovery from text in diverse fields and contexts.

REFERENCES

1. Books:
    - "R for Data Science" by Hadley Wickham and Garrett Grolemund.
2. Online Tutorials and Documentation:
    - R Project's official website : https://www.r-project.org/
    - W3schools : https://www.w3schools.com/
    - Geeksforgeeks : https://practice.geeksforgeeks.org/
3. Forums and Q&A:
    - Stack Overflow : https://stackoverflow.com/
    - RStudio Community : https://community.rstudio.com/