##Peer Graded Assignment Milestone. **Shivangi Mehta Date–11/7/2021**

Introduction

This is the project of Swiftkey company who invent digital Keys on Cell Phone. The purpose of this project is to identify the most frequent words used while texting using mobile phone. We are going to analyis this by using Blogs, twitter, News data set provided by Coursera in Data Science Specialization.

TASK 1

Getting and Cleaning Data

Tasks to accomplish

TASK 1.1## Tokenization - identifying appropriate tokens such as words, punctuation, and numbers. Writing a function that takes a file as input and returns a tokenized version of it.

TASK 1.2## Profanity filtering - removing profanity and other words you do not want to predict.

Criteria

1.The link lead to an HTML page describing the exploratory analysis of the training data set.

2.summaries of the three files. Word counts, line counts and basic data tables.

3.The data scientist made basic plots, such as histograms to illustrate features of the data.

4.The report written in a brief.

load All Require packages

{r Final Milestone,warning=FALSE} library(dplyr) library(ggplot2) require(readtext) library(sqldf) library(stringi) library(quanteda) **Load the Data**

{r, warning=FALSE} if(!file.exists("./data")){dir.create("./data")} fileUrl <- "https://d396qusza40orc.
download.file(fileUrl,destfile="./data/Coursera-SwiftKey.zip") **unzipped the file**

{r, warning=FALSE} unzip <- unzip(zipfile="./data/Coursera-SwiftKey.zip",exdir="./data")
unzip **Checking Data Connection by creating Summary**

```
con <- file("./data/final/en_US/en_US.twitter.txt", "r")
twittertxt <- readLines(con,encoding = "UTF-8", skipNul = TRUE )
con1 <- file("./data/final/en_US/en_US.blogs.txt", "r")
blogstxt <- readLines(con1, encoding = "UTF-8", skipNul = TRUE)
con2 <- file("./data/final/en_US/en_US.news.txt", "r")
newstxt <- readLines(con, encoding = "UTF-8", skipNul = TRUE)
close(con)
```

Checking number of lines, words and Size of all Three text file

```
sizeTwitter <- file.info("./data/final/en_US/en_US.twitter.txt")$size / 1024
sizeblogs <- file.info("./data/final/en_US/en_US.blogs.txt")$size / 1024
sizenews <- file.info("./data/final/en_US/en_US.news.txt")$size / 1024
wordsTwitter <- stri_count_words("./data/final/en_US/en_US.twitter.txt")
wordsblogs <- stri_count_words("./data/final/en_US/en_US.blogs.txt")
wordsnews <- stri_count_words("./data/final/en_US/en_US.news.txt")
mergedfile.txt <- data.frame(Source = c("Twitter", "News", "Blogs"),
           SizeMB = c(sizeTwitter,sizeblogs,sizenews),
           NoofLines = c(length(twittertxt),length(blogstxt), length(newstxt)),
           NoOfWords = c(sum(wordsTwitter),sum(wordsblogs), sum(wordsnews)))
```

Since this is the large set of data, this is difficult for us to load all of those within limited RAM memory in personal computer. Therefore, I am collecting Sample out of it as below to complete my analysis.

Sampling

```
set.seed(1288)
twitter_sample <- sample(twittertxt, length(twittertxt)*0.01, replace = FALSE)
blogs_sample <- sample(blogstxt, length(blogstxt)*0.01, replace = FALSE)
news_sample <- sample(newstxt, length(newstxt)*0.01, replace = FALSE)
Sample_data <- c(twitter_sample, blogs_sample, news_sample)
```

The new dataset consists of 32593 words.

TASK 2

I am removing those Text which I don't want to analyis.

##Clean Data more in Dept.

```
##Detail Cleaning.We are using gsub() function to remove all unwanted symbols from the data.44

Sample_data <- gsub("&amp", "", Sample_data)

# remove tweets
Sample_data <- gsub("RT :|@[a-z,A-Z]*: ", "", Sample_data)
Sample_data <- gsub("@\\w+", "", Sample_data)

# remove digits
Sample_data <- gsub("[[:digit:]]", "", Sample_data)

# remove hash tags
Sample_data <- gsub(" #\\S*","", Sample_data)

# remove url
Sample_data <- gsub(" ?(f|ht)tp(s?)://(.*)[.][a-z]+", "", Sample_data)

# Remove NA NA

Sample_data <- gsub("NA NA", " ", Sample_data)

# remove extra spaces

library(qdapRegex)
Sample_data <- rm_white(Sample_data)
```

In the above R chunks we have cleaned data Sample. We are now ready for Creating Corpus.

We want to determined and experiment with 3 different grams. Unigram, Bigram and Trigram. We will check words usage frequency and then will create mathematical diagram to check the words frequency and compare all of the three text file.

**Here is the data frame from the Sample data for n-gram.

{r, warning=FALSE} data_sample_df <- tibble::tibble(line = 1:length(Sample_data), text = Sample_data)

**1. Creating and Showing Unigram.

“‘{r, warning=FALSE} UnigramFreq <- data_sample_df %>% unnest_tokens(unigram, text, token = “ngrams”, n = 3) %>% separate(unigram, c(“word1”), sep = " “, extra =”drop“, fill =”right“) %>% filter(!word1 %in% stop_words$word) %>% unite(unigram, word1, sep =” ") %>% count(unigram, sort = TRUE)

ggplot(head(UnigramFreq,15), aes(reorder(unigram,n), n)) +
geom_bar(stat=“identity”) + coord_flip() + xlab(“Unigrams”) + ylab(“Frequency”) + ggtitle(“Frequently used Unigrams”)

```
**2. Plotting Bigram words frequency data**.

‘‘‘{r, warning=FALSE}
BigramFreq <- data_sample_df %>%
    unnest_tokens(bigram, text, token = "ngrams", n = 3) %>%
    separate(bigram, c("word1", "word2"), sep = " ",
             extra = "drop", fill = "right") %>%
    filter(!word1 %in% stop_words$word,
           !word2 %in% stop_words$word) %>%
    unite(bigram, word1, word2, sep = " ") %>%
    count(bigram, sort = TRUE)

ggplot(head(BigramFreq,15), aes(reorder(bigram,n), n)) +
    geom_bar(stat="identity") + coord_flip() +
    xlab("Bigrams") + ylab("Frequency") +
    ggtitle("Frequently used Bigrams")
```

3. Plotting Trigram words frequency data

“‘{r, warning=FALSE} TrigramFreq <- data_sample_df %>% unnest_tokens(trigram, text, token = “ngrams”, n = 3) %>% separate(trigram, c(“word1”, “word2”, “word3”), sep = " “, extra =”drop“, fill =”right“) %>% filter(!word1 %in% stop_words*word*, !*word*2word, !word3 %in% stop_words$word) %>% unite(trigram, word1, word2, word3, sep =” ") %>% count(trigram, sort = TRUE)

ggplot(head(TrigramFreq,15), aes(reorder(trigram,n), n)) +
geom_bar(stat=“identity”) + coord_flip() + xlab(“Trigrams”) + ylab(“Frequency”) + ggtitle(“Frequently used Trigrams”) “‘

Report from the Data Analysis

In this project, I have downloaded data from the coursera. I have install all the require packages. Instead of corpus I used gsub() since, it took longer time to run Corpus. By using ngrams method, I have create plots for Unigram, Bigram and Trigram.I have removed stopwords to get clear information about words frequency.

For more detial I will use Shiny application to present correct analysis on this project.