Assignment 2

Rohan Chouthai

May 22, 2018

Let us first load all the libraries. I have already installed these packages previously for different works in R and hence I would only be calling the libraries at this moment.

```
library(tibble)
```

```
## Warning: package 'tibble' was built under R version 3.4.3
```

```
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 3.4.3
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 3.4.4
```

```
library(udpipe)
```

```
## Warning: package 'udpipe' was built under R version 3.4.4
```

SECTION 1: Data Cleaning and Exploration

Now, let us load the Hindi dataset in R. This dataset is a Hindi news corpus taken from http://wortschatz.uni-leipzig.de/en/download/ (http://wortschatz.uni-leipzig.de/en/download/). It basically comprises of Hindi news from 2011.

I will read it as a table since it will be easier for the analysis.

```
hindi_news<-read.table("C:/Users/rohan/Desktop/Data Science with Python/docs/hin_news_2011_100K-
sentences.txt",header = FALSE,sep = "\n",stringsAsFactors = F)
```

```
## Warning in scan(file = file, what = what, sep = sep, quote = quote, dec =
## dec, : EOF within quoted string
```

Section 2: Staging with pretrained language model

I will first download the hindi model.

```
library(udpipe)
model_hin <- udpipe_download_model(language = "hindi")
```

```
## Downloading udpipe model from https://raw.githubusercontent.com/jwijffels/udpipe.models.ud.2.
0/master/inst/udpipe-ud-2.0-170801/hindi-ud-2.0-170801.udpipe to C:/Users/rohan/Desktop/Data Sci
ence with Python/hindi-ud-2.0-170801.udpipe
```

Since I am running the hindi model for the first time, I will execute the following line of code.

```
udmodel_hindi <- udpipe_load_model(file ="C:/Users/rohan/Downloads/hindi-ud-2.0-170801.udpipe" )
```

It is important to give the machine some intuition of the data so that the machine can trace patterns in the data and make out better insights from it. Thus, accurate metadata should be added to the dataset. This process of assigning metadata is called annotation. We will do the annotation using udpipe.

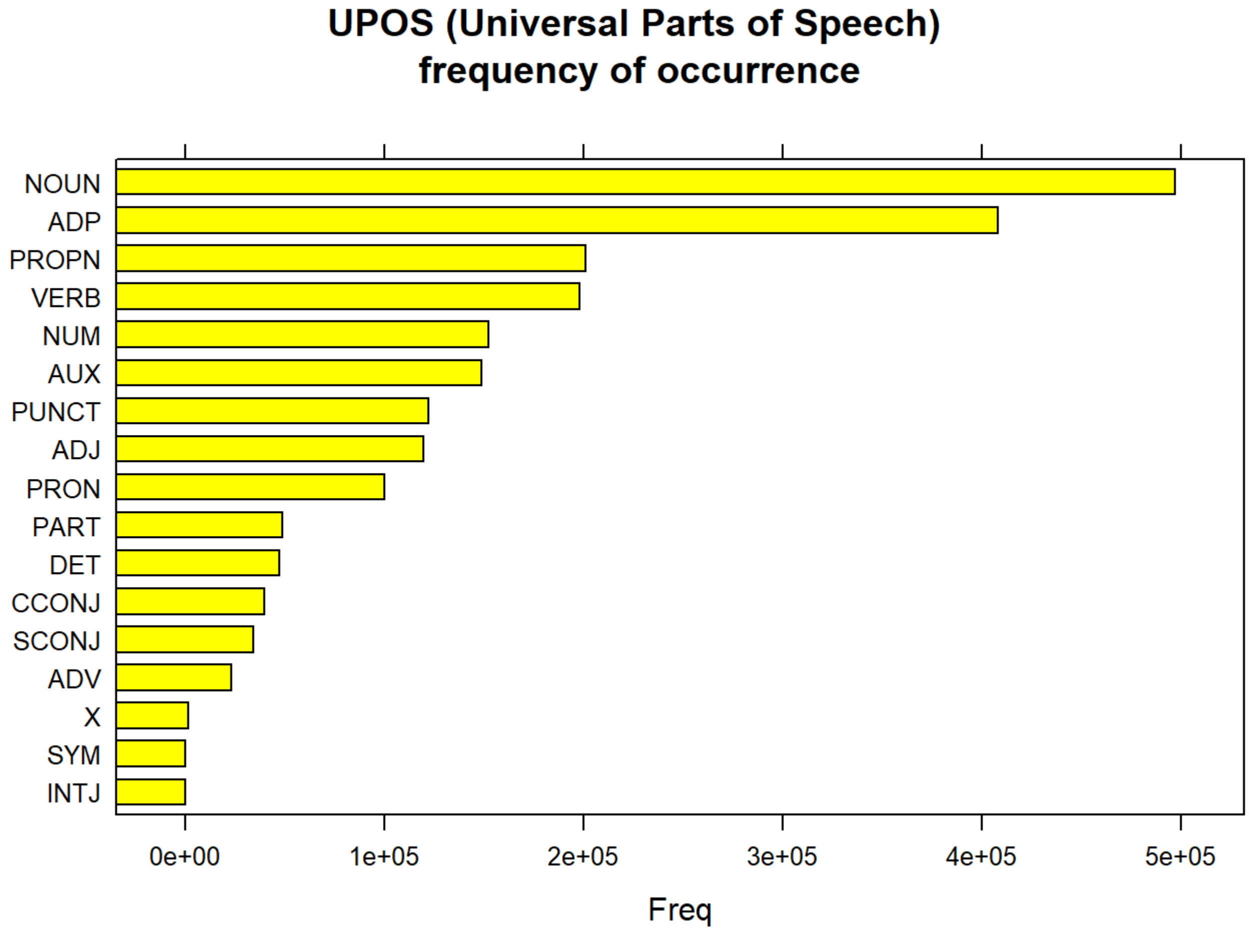
Udpipe annotate() takes the language model and annotates the given text data.

```
p <- udpipe_annotate(udmodel_hindi, hindi_news$V1)
q <- data.frame(p)
```

Section 3: Analysis

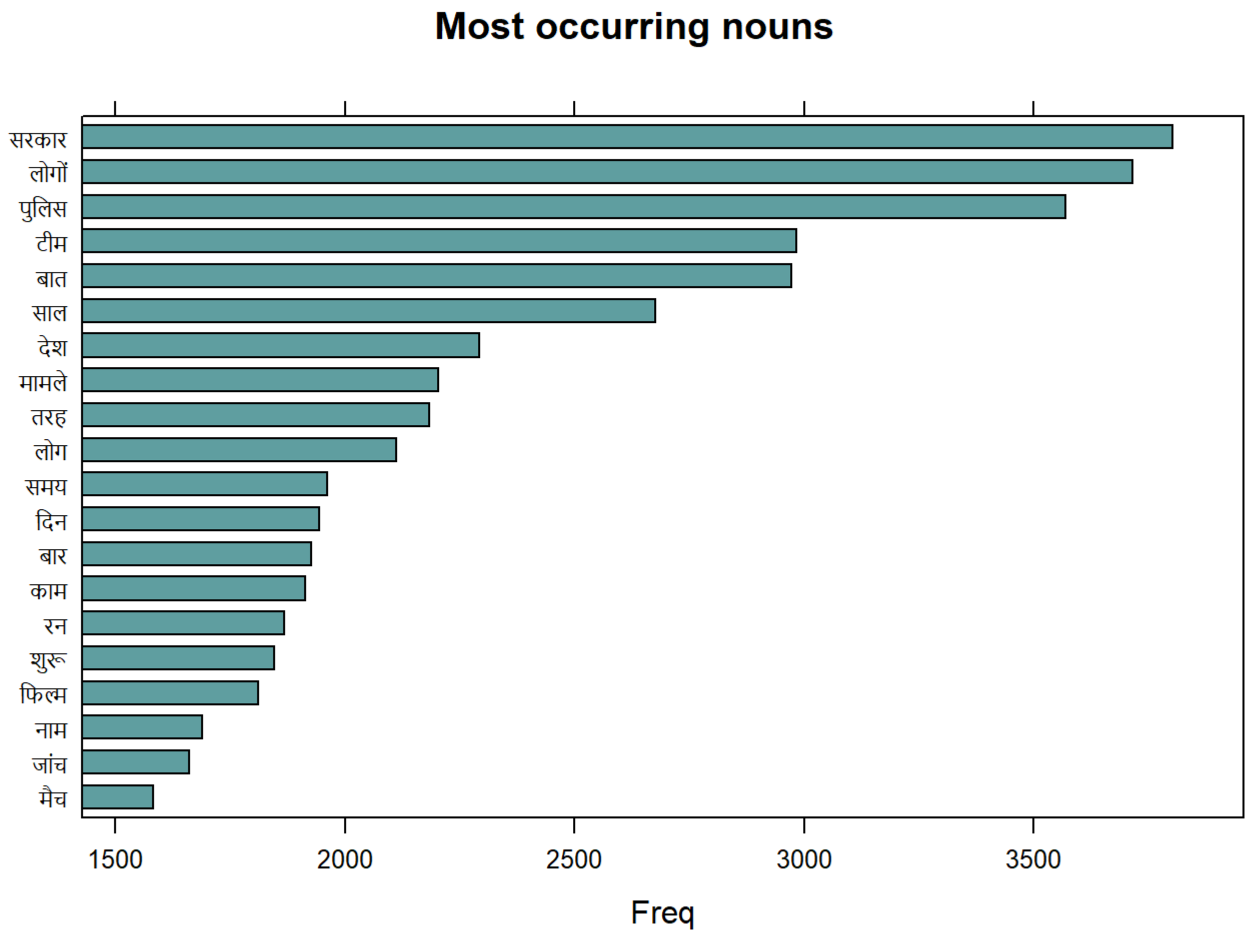
I will use the lattice package to plot the parts of speech from my hindi text.

```
library(lattice)
stats <- txt_freq(q$upos)
stats$key <- factor(stats$key, levels = rev(stats$key))
barchart(key ~ freq, data = stats, col = "yellow",
         main = "UPOS (Universal Parts of Speech)\n frequency of occurrence",
         xlab = "Freq")
```



It can be seen that lot of them are nouns. Since we have got the parts of speech annotated, we can try to tally this up.

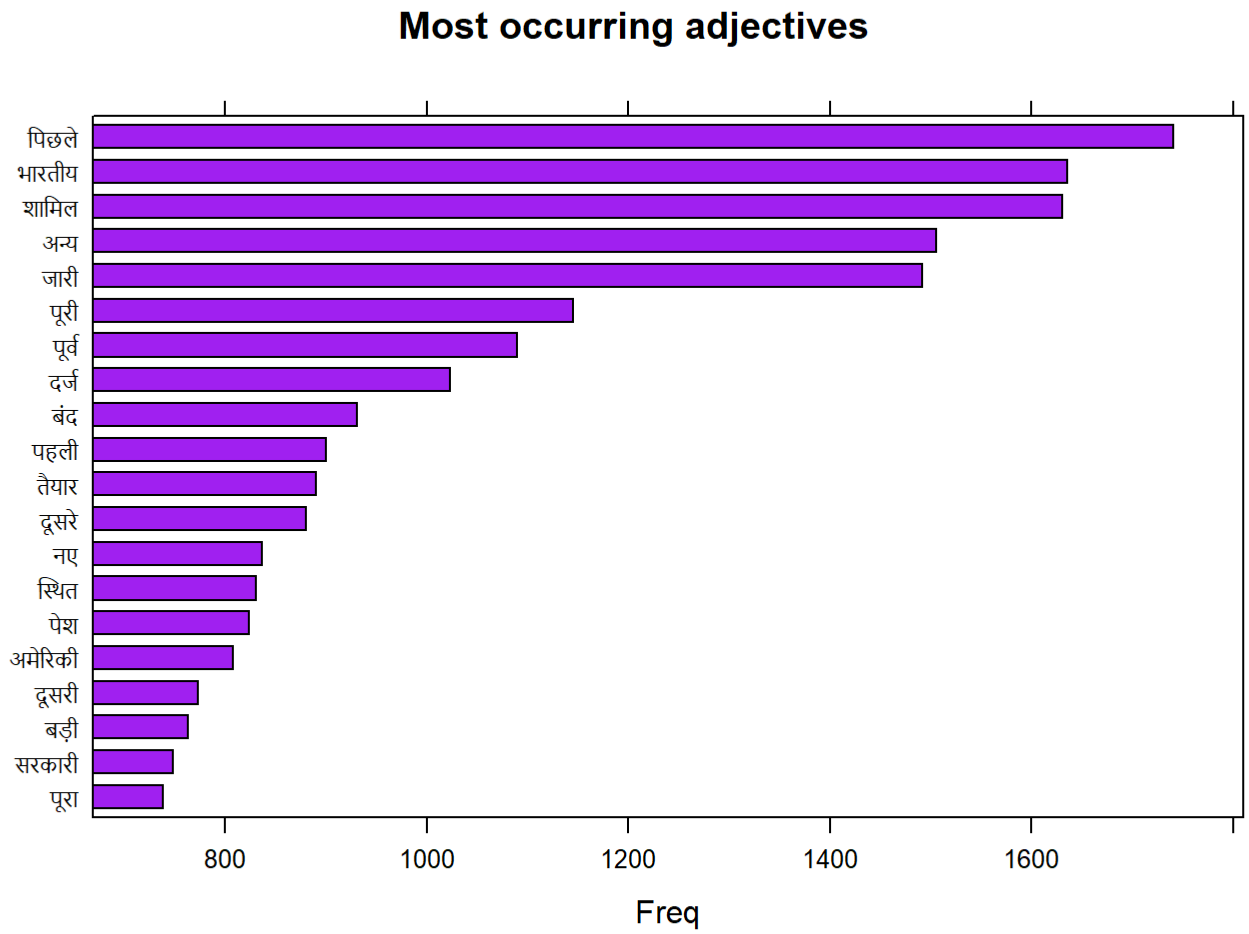
```
stats <- subset(q, upos %in% c("NOUN"))
stats <- txt_freq(stats$token)
stats$key <- factor(stats$key, levels = rev(stats$key))
barchart(key ~ freq, data = head(stats, 20), col = "cadetblue",
         main = "Most occurring nouns", xlab = "Freq")
```



We can see that the most occuring noun is Government followed by people and police. It seems like most news were centered around these nouns in 2011.

Let us also explore how adjectives fare since like English, Hindi also uses a lot of expressive adjectives.

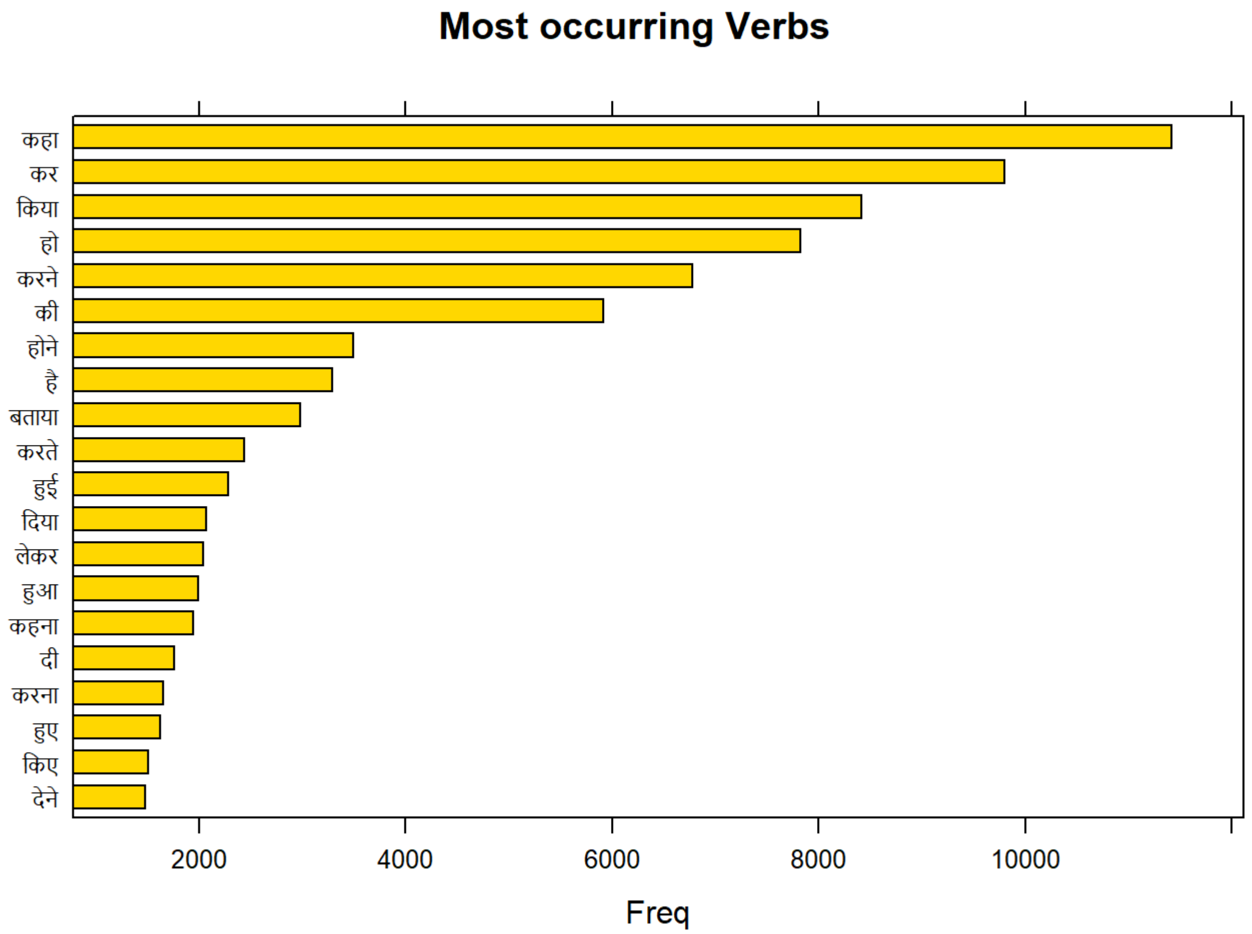
```
stats <- subset(q, upos %in% c("ADJ"))
stats <- txt_freq(stats$token)
stats$key <- factor(stats$key, levels = rev(stats$key))
barchart(key ~ freq, data = head(stats, 20), col = "purple",
         main = "Most occurring adjectives", xlab = "Freq")
```



We can see that the adjective Indian features a lot. The news corpus being analyzed seems to have talked a lot about Indian news and Indian government a lot. Most of the talk seems to have been centered about what happened in India more than anything else.

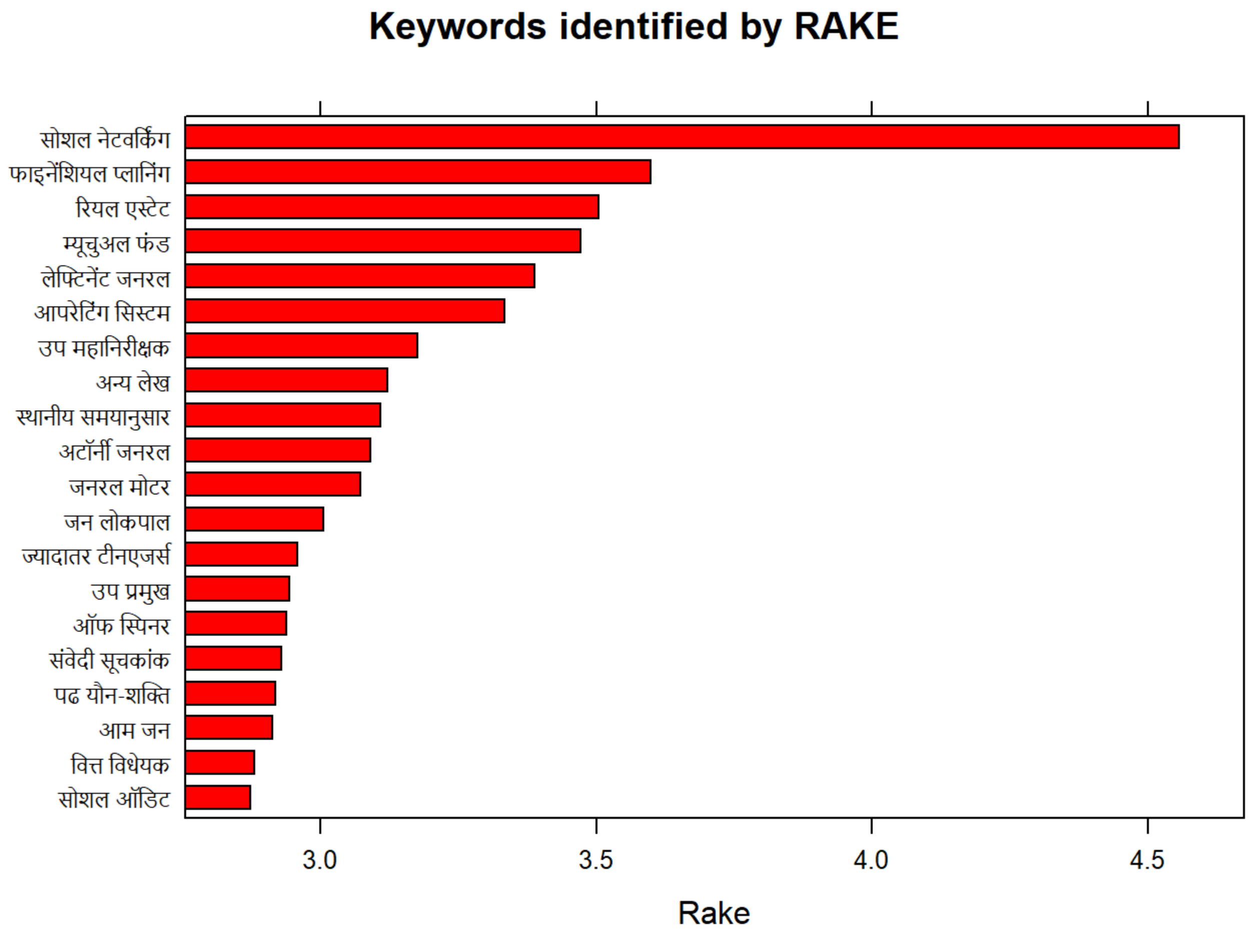
It is often worthwhile checking the verbs as well because they give a good idea of the general sentiment infused- whether it is a positive or optimistic one or a negative or pessimistic one, Let us check.

```
stats <- subset(q, upos %in% c("VERB"))
stats <- txt_freq(stats$token)
stats$key <- factor(stats$key, levels = rev(stats$key))
barchart(key ~ freq, data = head(stats, 20), col = "gold",
         main = "Most occurring Verbs", xlab = "Freq")
```



RAKE is one of the most popular unsupervised algorithms for extracting keywords in Information retrieval. RAKE is short for Rapid Automatic Keyword Extraction. It is a domain independent algorithm which tries to determine key phrases in a body of text by analyzing the frequency of word appearance as well as its co-occurrence with other words in the corpus Let's goup nouns and adjectives for a better understanding of the roles of nouns,

```
stats <- keywords_rake(x = q, term = "lemma", group = "doc_id",
                       relevant = q$upos %in% c("NOUN", "ADJ"))
stats$key <- factor(stats$keyword, levels = rev(stats$keyword))
barchart(key ~ rake, data = head(subset(stats, freq > 3), 20), col = "red",
         main = "Keywords identified by RAKE",
         xlab = "Rake")
```



We can see that the top keywords are social networking, financial planning, real estate and mutual funds. These keywords

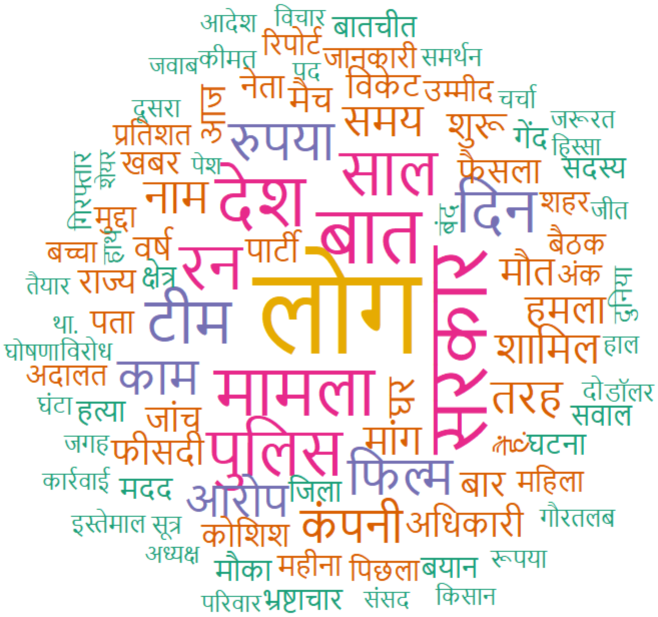
Building wordcloud helps get a fancy view of the most occuring words in the text. Let us go ahead and make a wordcloud.

```
library(wordcloud)
```

```
## Warning: package 'wordcloud' was built under R version 3.4.3
```

```
## Loading required package: RColorBrewer
```

```
wordcloud(words = stats$key, freq = stats$freq, min.freq = 3, max.words = 100,
          random.order = FALSE, colors = brewer.pal(6, "Dark2"))
```



Part 5- Data Science Conclusion In this analysis, the aim is to get a sense of how the Hindi news landscape looked like back in India in 2011. The RAKE word frequency goes on to show the spike in social media and the traction it gained in the news back in 2011. Facebook and Twitter picked up the most during this time and social media came to be used not only as a pass time but an effective tool for targeted advertisement. The financial planning and mutual funds keywords indicate that Indian public back in 2011 was focused quite a lot on how they plan their finances and mutual funds seem to have been much discussed as well.

The word cloud shows that a lot of news was centered around ‘people’. This reaffirms the common notion of India being a people’s democracy. The government also featured in the news a lot and so did police. Surprisingly, a lot of the news also included the word for accusation. This indicates a widespread and rampant level of controversies which happen in India over a variety of issues. Being a diverse country, people are very passionate about their beliefs and often resort to wild accusations which turn into long standing controversies. The words for runs and films also show two of the biggest things Indians are passionate about- cricket and bollywood.

In summary, we can see how important and effective natural language processing can be even when it is yielded towards a language such as Hindi which is not as commonly spoken as the likes of English.