FDS_Assignment_TextMining_For_WordCloud

Prabhat Ale

2024-04-30

Loading libraries for doing text mining and scraping contents from the web

```
library(readr)
library(tm)
```

```
## Loading required package: NLP
```

```
library(rvest)
```

```
##
## Attaching package: 'rvest'
```

```
## The following object is masked from 'package:readr':
##
##     guess_encoding
```

This is the url from where we are going to scrape the contents for creating a word cloud.

```
url <- "https://thehimalayantimes.com/opinion/navigating-nepals-digital-frontier-understanding-cybersecu
```

Reading html contents from the url

```
data <- read_html(url)
```

Extracting the relevant information from the 'post-content' div class

```
opinions <- data %>% html_element('.post-content')
```

Gathering the opinions from the paragraph nodes within a html text.

```
final_opinions <- opinions %>% html_nodes('p') %>% html_text()
```

creating a text corpus from the himalyantimes news paragraph

```
corpus <- Corpus(VectorSource(final_opinions))
```

lowercasing the text

```
corpus <- tm_map(corpus, tolower)
```

```
## Warning in tm_map.SimpleCorpus(corpus, tolower): transformation drops documents
```

inspecting the top 3 documents

```r
inspect(corpus[1:3])
```

```
## <<SimpleCorpus>>
## Metadata:  corpus specific: 1, document level (indexed): 0
## Content:  documents: 3
##
## [1] in recent years, nepal has made momentous advances in digital space. less than 10 percent of the
## [2] as the country enters the world of digits, every aspect of life in the digital sector becomes dat
## [3] building a security framework
```

removing punctuations

```r
corpus <- tm_map(corpus, removePunctuation)
```

```
## Warning in tm_map.SimpleCorpus(corpus, removePunctuation): transformation drops
## documents
```

inspecting the top 3 documents after removing punctuations

```r
inspect(corpus[1:3])
```

```
## <<SimpleCorpus>>
## Metadata:  corpus specific: 1, document level (indexed): 0
## Content:  documents: 3
##
## [1] in recent years nepal has made momentous advances in digital space less than 10 percent of the po
## [2] as the country enters the world of digits every aspect of life in the digital sector becomes data
## [3] building a security framework
```

removing numbers

```r
corpus <- tm_map(corpus, removeNumbers)
```

```
## Warning in tm_map.SimpleCorpus(corpus, removeNumbers): transformation drops
## documents
```

inspecting the top 3 documents after removing numerical values

```r
inspect(corpus[1:3])
```

```
## <<SimpleCorpus>>
## Metadata:  corpus specific: 1, document level (indexed): 0
## Content:  documents: 3
##
## [1] in recent years nepal has made momentous advances in digital space less than  percent of the popu
## [2] as the country enters the world of digits every aspect of life in the digital sector becomes data
## [3] building a security framework
```

creating a function to remove urls from text

```r
remove_url <- function(x) gsub ('http[^[:space:]]*', "", x)
```

calling a function to remove urls from documents

```r
corpus <- tm_map(corpus, remove_url)
```

```
## Warning in tm_map.SimpleCorpus(corpus, remove_url): transformation drops
## documents
```

creating a function to remove new line characters

```r
remove_newline_chars <- function(x) gsub ('\n', '', x)
```

calling a function to remove new line characters from documents

```r
corpus <- tm_map(corpus, remove_newline_chars)
```

```
## Warning in tm_map.SimpleCorpus(corpus, remove_newline_chars): transformation
## drops documents
```

creating a function to replace multiple spaces with a single space

```r
removeMultipleSpaces <- function(x) gsub('\\s+', ' ', x)
```

calling a function to remove multiple spaces with a single space

```r
corpus <- tm_map(corpus, removeMultipleSpaces)
```

```
## Warning in tm_map.SimpleCorpus(corpus, removeMultipleSpaces): transformation
## drops documents
```

Removing stopwords

```r
clean_corpus <- tm_map(corpus, removeWords, stopwords("english"))
```

```
## Warning in tm_map.SimpleCorpus(corpus, removeWords, stopwords("english")):
## transformation drops documents
```

Creating a summary of a clean corpus

```r
summary(clean_corpus)
```

```
##    Length Class             Mode
## 1  2      PlainTextDocument list
## 2  2      PlainTextDocument list
## 3  2      PlainTextDocument list
## 4  2      PlainTextDocument list
## 5  2      PlainTextDocument list
## 6  2      PlainTextDocument list
## 7  2      PlainTextDocument list
## 8  2      PlainTextDocument list
## 9  2      PlainTextDocument list
## 10 2      PlainTextDocument list
## 11 2      PlainTextDocument list
## 12 2      PlainTextDocument list
## 13 2      PlainTextDocument list
## 14 2      PlainTextDocument list
## 15 2      PlainTextDocument list
## 16 2      PlainTextDocument list
## 17 2      PlainTextDocument list
## 18 2      PlainTextDocument list
## 19 2      PlainTextDocument list
## 20 2      PlainTextDocument list
## 21 2      PlainTextDocument list
```

Creating a document term matrix

```r
dtm <- DocumentTermMatrix(clean_corpus)
```

finding frequent terms having frequency greater than or equal to 2

```r
(freq_terms <- findFreqTerms(dtm, lowfreq = 2))
```

```
##  [1] "digital"       "internet"      "nepal"         "nepali"       
##  [5] "percent"       "population"    "space"         "authorities"  
##  [9] "country"       "data"          "every"         "government"   
## [13] "governments"   "important"     "including"     "individuals"  
## [17] "information"   "life"          "organizations" "sector"       
## [21] "security"      "world"         "building"      "framework"    
## [25] "approach"      "build"         "business"      "cybersecurity"
## [29] "robust"        "significant"   "time"          "towards"      
## [33] "along"         "assets"        "can"           "plans"        
## [37] "policies"      "set"           "systems"       "technologies" 
## [41] "threats"       "april"         "intelligence"  "microsoft"    
## [45] "threat"        "citizens"      "cyber"         "economic"     
## [49] "ensure"        "growth"        "investments"   "nation"       
## [53] "system"        "trust"         "also"          "collaboration"
## [57] "detection"     "different"     "training"      "leadership"   
## [61] "become"        "direction"     "handling"      "often"        
## [65] "practices"     "protect"       "smart"         "strategic"    
## [69] "technology"    "zero"          "develop"       "employees"    
## [73] "potential"     "investment"    "journey"       "multiyear"    
## [77] "development"   "global"        "prosperity"    "ais"
```

correlated terms with country

```r
findAssocs(dtm, 'country', 0.6)
```

```
## $country
## authorities        life      sector  government       every 
##        0.79        0.79        0.79        0.75        0.61
```

Loading a library to create a word cloud

```
library(wordcloud)
```

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

Creating a document term matrix of the words represented in a corpus

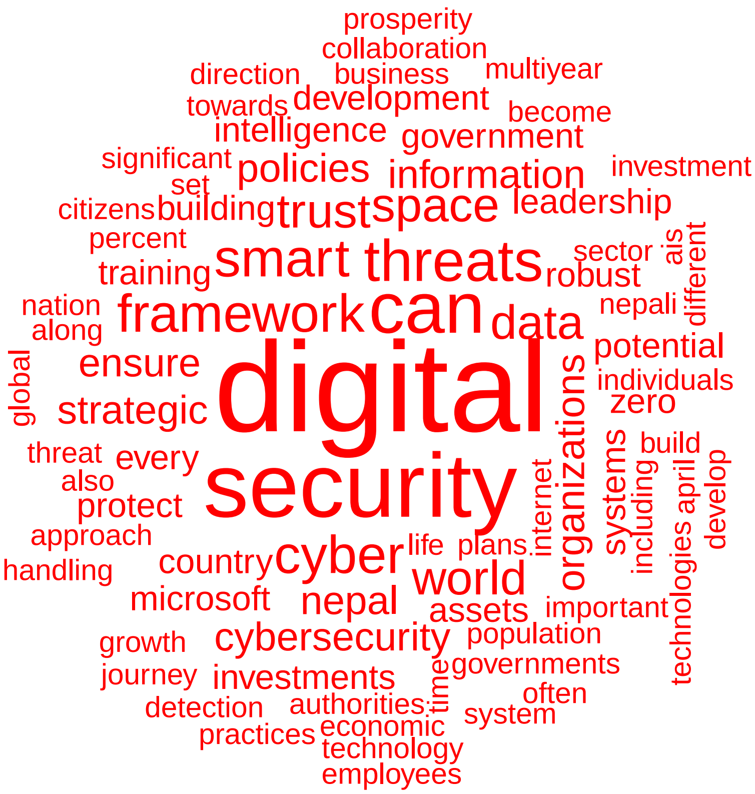
```
document_matrix <- as.matrix(dtm)
```

Creating a word frequency for each terms in all the documents

```
word_freq <- sort(colSums(document_matrix), decreasing = T)
```

Creating a word cloud using the provided word frequency

```
wordcloud(word = names(word_freq), freq = word_freq, min.freq = 2, random.order = F, colors = 'red')
```


prosperity
collaboration
direction
business
multiyear
towards
development
become
intelligence
government
significant
policies
information
investment
set
citizens
building
trust
space
leadership
percent
smart
threats
sector
ais
different
training
robust
nation
nepali
along
framework
can
data
potential
global
ensure
digital
individuals
zero
strategic
organizations
build
threat
every
systems
internet
including
april
develop
also
security
protect
approach
life
plans
technologies
handling
country
cyber
world
microsoft
nepal
assets
important
growth
cybersecurity
population
journey
investments
time
governments
detection
authorities
often
system
practices
economic
technology
employees