Scrapping from Reddit

Rodrigo Esteves de Lima-Lopes
State University of Campinas
rll307@unicamp.br

Contents

1 Packages **1**

2 Searching for a theme in Reddit **1**

3 Downloading most commented link **1**
3.1 Second possibility 2

4 Analysis **2**
4.1 Selecting some information 2

5 Plotting **4**
5.1 Network 4
5.2 Information plotting 5

1 Packages

```
library(RedditExtractoR)
library(dplyr)
library(igraph)
library(tidygraph)
library(ggraph)
library(visNetwork)
```

2 Searching for a theme in Reddit

```
links <- reddit_urls(
  search_terms   = "coronavac",
  page_threshold = 5
  ) %>%
  mutate(title=iconv(title, from = "latin1", to = "UTF-8"))%>%
  mutate(URL=iconv(URL, from = "latin1", to = "UTF-8"))
```

3 Downloading most commented link

```
## First possibility
cornoavac.1 <- reddit_content("https://www.reddit.com/r/brasil/comments/kse8x4/coronavac_tem_efic%C3%A1
```

```
  mutate(comment=iconv(comment, from = "latin1", to = "UTF-8"))%>%
  mutate(title=iconv(title, from = "latin1", to = "UTF-8"))
```

3.1 Second possibility

```
cornoavac.2 <- get_reddit(subreddit = 'brasil',
                  search_terms= 'coronavac',
                  cn_threshold=20,
                  page_threshold=15) %>%
  mutate(comment=iconv(comment, from = "latin1", to = "UTF-8"))%>%
  mutate(title=iconv(title, from = "latin1", to = "UTF-8"))
```

4 Analysis

4.1 Selecting some information

```
covid.posts <-cornoavac.2
```

Making the data as they should:

```
covid.posts$post_date <- as.Date(covid.posts$post_date, '%d-%m-%y')
```

Selecting comments per day

```
comments.day <-select(cornoavac.2, comm_date)
comments.day<- data.frame(table(comments.day))
```

1. Deleting the factor feature
2. Making dates as dates
3. Making numbers as numbers

```
str(comments.day)
comments.day  <- data.frame(lapply(comments.day, as.character), stringsAsFactors=FALSE)
comments.day$comments.day<-as.Date(comments.day$comments.day,'%d-%m-%y')
colnames(comments.day)<-c('day','value')
comments.day$value<-as.numeric(comments.day$value)
```

Selecting post per days:

```
post.day <- select(covid.posts, post_date)
post.day <- data.frame(table(post.day))
post.day <- data.frame(lapply(post.day, as.character), stringsAsFactors=FALSE)
post.day$post.day<-as.Date(post.day$post.day, ' %Y-%m-%d')
post.day$Freq<-as.numeric(post.day$Freq)
colnames(post.day)<-c('day','value')
str(post.day)
mean(post.day$value)
```

First analysis of users network

```
my.post<-subset(covid.posts,post_date == "2020-11-13" & num_comments > 70)
users <- user_network(my.post,include_author = TRUE, agg = FALSE)
users$plot

nodes <-as.data.frame(users[["nodes"]])
edges <-as.data.frame(users[["edges"]])
```

```
edges$title<-NULL
#Get the nodes anonymous
nodes$user.id<-paste0(rep('u',43),1:43)
```

Nodes degrees

```
nodes2 <- nodes
nodes2$user<-NULL
colnames(nodes2)<-c("id","user")

table.nodesf<-data.frame(table(select(edges, 'from')))
table.nodest<-data.frame(table(select(edges, 'to')))
colnames(table.nodest)<-c('id','to')
colnames(table.nodesf)<-c('id','from')
table.nodes<-merge(table.nodesf,table.nodest,all=T)
table.nodes[is.na(table.nodes)]<-0
table.nodes$total<-rowSums(table.nodes[2:3])
nodes2$value<-table.nodes$total
```

New edges

```
edges2<- data.frame(edges)
colnames(edges2)<-c('from','to','value')
```

font size

```
nodes2 <- nodes2 %>%
  mutate(font.size = 30)
nodes2$group<-rep("C0",43)
```

Defining groups

```
nodes2[43,5]<-"Author"

# Community 1
nodes2[38,5]<-"C1"
nodes2[20,5]<-"C1"
nodes2[21,5]<-"C1"
nodes2[42,5]<-"C1"
nodes2[40,5]<-"C1"
nodes2[41,5]<-"C1"

#Bridges
nodes2[3,5]<-"B"
nodes2[8,5]<-"B"
nodes2[16,5]<-"B"
nodes2[18,5]<-"B"
nodes2[28,5]<-"B"#
nodes2[30,5]<-"B"
nodes2[38,5]<-"B"

# Community 2
nodes2[19,5]<-"C2"

# Community 3
nodes2[10,5]<-"C3"
```

```
nodes2[11,5]<-"C3"
nodes2[12,5]<-"C3"
nodes2[13,5]<-"C3"

# Community 4
nodes2[5,5]<-"C4"
nodes2[6,5]<-"C4"

# Community 5
nodes2[31,5]<-"C5"

#Community 6
nodes2[29,5]<-"C6"
```

5 Plotting

5.1 Network

```
nodes2$label<-nodes2$user
nodes2[43,6]<-"Author"
```

5.1.1 G1

Creating the graph

```
routes_tidy <- igraph::graph_from_data_frame(edges2, vertices = nodes2) %>%
  as_tbl_graph()
```

Plotting

```
ggraph(routes_tidy, layout = "fr") +
  geom_node_point() +
  geom_edge_link(aes(width = value), alpha = 0.8) +
  scale_edge_width(range = c(0.2, 2)) +
  geom_node_text(aes(label = user), repel = TRUE) +
  labs(edge_width = "Connections") +
  theme_graph()
```

5.1.2 G2

```
visNetwork(nodes2, edges2, physics=T)%>%
  visEdges(arrows = "to")
```

Using groups

```
visNetwork(nodes2, edges2)%>%
  visEdges(arrows =list(to=list(enabled = TRUE, scaleFactor = 2)))%>%
  visGroups(groupname = "C0") %>%
  visGroups(groupname = "Author", color = "brown",shape = "box",
            shadow = list(enabled = TRUE),size=45) %>%
  visGroups(groupname = "B", color = "magenta",shape = "ellipse",
            size = 30) %>%
  visGroups(groupname = "C1", color = "purple") %>%
  visGroups(groupname = "C2", color = "green") %>%
```

```
  visGroups(groupname = "C3", color = "yellow") %>%
  visGroups(groupname = "C4", color = "darkyellow") %>%
  visGroups(groupname = "C5", color = "cyan") %>%
  visGroups(groupname = "C5", color = "black") %>%
  visPhysics(solver = "forceAtlas2Based", forceAtlas2Based= list(avoidOverlap=
                                                                   )
             )
```

Hierarchical Layout

```
visNetwork(nodes2, edges2) %>% visHierarchicalLayout(edgeMinimization=F,nodeSpacing=200)
```

5.2 Information plotting

Scores vs votes

```
covid.posts%>%
  ggplot(., aes(y=post_score,x=upvote_prop))+
  geom_point(alpha = 1/3) +
  geom_hline(yintercept = mean(covid.posts$post_score), color="red",linetype=5,size=1.3)+
  geom_vline(xintercept = mean(covid.posts$upvote_prop), color="steelblue",linetype=5,size=1.3)+
  annotate(geom="text", label="Post Score Avarage", y=mean(covid.posts$post_score), x=0.96, vjust=-1,co
  annotate(geom="text", label="Post Upvote Avarage", x=0.96, y=1000, vjust=-1,color='steelblue')
```

A Complex graph

```
covid.posts%>%
  ggplot(., aes(y=num_comments,x=comment_score))+
  geom_point(aes(size=controversiality),shape = 21,colour = "black", fill = "white", stroke = 1)+
  geom_hline(yintercept = mean(covid.posts$num_comments), color="red",linetype="solid",size=1.5)+
  geom_vline(xintercept = mean(covid.posts$comment_score), color="coral4",linetype="solid",size=1.5)+
  annotate(geom="text", label="Comment Average", y=mean(covid.posts$num_comments), x=60, vjust=-1,color
  annotate(geom="text", label="Comment Score Avarage", x=(mean(covid.posts$comment_score)+25), y=77, vj
```

Mean of comments

```
comments.day%>%
  ggplot(., aes(x=day, y=value)) +
  geom_line(color="steelblue") +
  geom_hline(yintercept = mean(comments.day$value), color="red",linetype="dashed",size=1.0)+
  geom_point(aes(size=value))+
  geom_label(data=subset(comments.day, value > 50),
             aes(label=day))+
  xlab("") +
  theme(axis.text.x=element_text(angle=60, hjust=1)) +
  scale_x_date(date_labels = "%e %b", date_breaks = "week",
               limit=c(as.Date("2020-11-10"),as.Date("2021-01-09")))+
  annotate(geom="text", label="Comment Average per day (33)", x=as.Date(1,origin = '2020-11-23'), y=30,
```

Posts per Day

```
post.day%>%
  ggplot(., aes(x=day, y=value)) +
  geom_line(color="steelblue") +
  geom_hline(yintercept = mean(post.day$value), color="red",linetype="dashed",size=1.0)+
  geom_point(aes(size=value))+
  geom_label(data=subset(post.day, value > 48),
```

```
            aes(label=day))+
  xlab("") +
  theme(axis.text.x=element_text(angle=60, hjust=1)) +
  scale_x_date(date_labels = "%e %b", date_breaks = "week",
               limit=c(as.Date("2020-11-10"),
                       as.Date("2021-01-10")))+
  annotate(geom="text", label="Avarage Posts per Day",
           x=as.Date(1,origin = '2020-11-10'), y=100,
           vjust=-1,color='red')
```

Comments per day

```
comments.day%>%
  ggplot(., aes(x=day, y=value)) +
  geom_line(color="steelblue") +
  geom_hline(yintercept = mean(comments.day$value), color="red",linetype="dashed",size=1.0)+
  geom_point(aes(size=value))+
  geom_label(data=subset(comments.day, value > 10),
            aes(label=day))+
  xlab("") +
  theme(axis.text.x=element_text(angle=60, hjust=1)) +
  scale_x_date(date_labels = "%e %b",
               date_breaks = "week",
               limit=c(as.Date("2020-11-10"),
                       as.Date("2021-01-10")))+
  annotate(geom="text", label="Comment Average per day (33)", x=as.Date(1,origin = '2020-11-23'),
           y=30, vjust=-1,color='red')
```