

Twitter zzh-gj-jz association

26 May, 2022

In this analysis I will first extract tweets mentioning “#zhangzhehan”, “#gongjun” or “#junzhe” and then check the overlap between accounts tweeting about these topics. After playing around a bit I decided to use only hashtags instead of full names to focus on dedicated tweets instead of casual mentions.

First let's get the tweets via Twitter API. Please note that there is a rate limit of up to 18000 tweets per day. Nonetheless, I never manage to get the maximum of 6000 tweets per query for some reason. Therefore this search is limited to the last couple of days.

I will then plot the accounts with the most tweets about these topics.

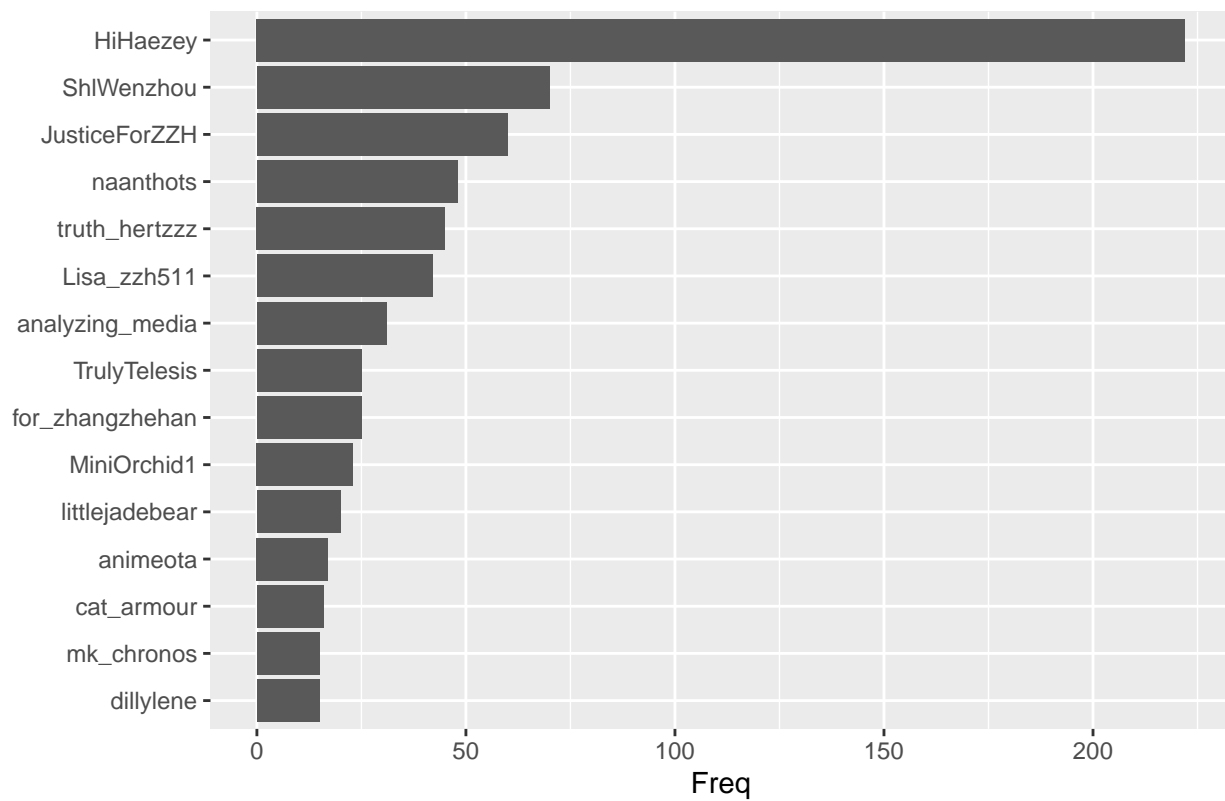
```
###extract tweets

tweets_extract_plot<-function(words,title){
  tweets<-search_tweets2(words,n = 6000,lang = "en",include_rts = FALSE)
  n_tweets<-nrow(tweets)
  stats<-paste(title,collapse=" ")
  stats<-paste(stats," n=",n_tweets,"tweets")

  plot<-table(tweets$screen_name)%>%
    as.data.frame() %>%
    arrange(desc(Freq)) %>%
    top_n(15) %>%
    mutate(Var1 = reorder(Var1, Freq)) %>%
    ggplot(aes(x = Var1, y = Freq)) +
    geom_col() +
    xlab(NULL) +
    coord_flip() +
    ggtitle(paste("top accounts tweeting about",stats))
  print(plot)
  return(tweets)
}
zzh_tweets<-tweets_extract_plot(c("#zhangzhehan"),"#zhangzhehan")
```

```
## Selecting by Freq
```

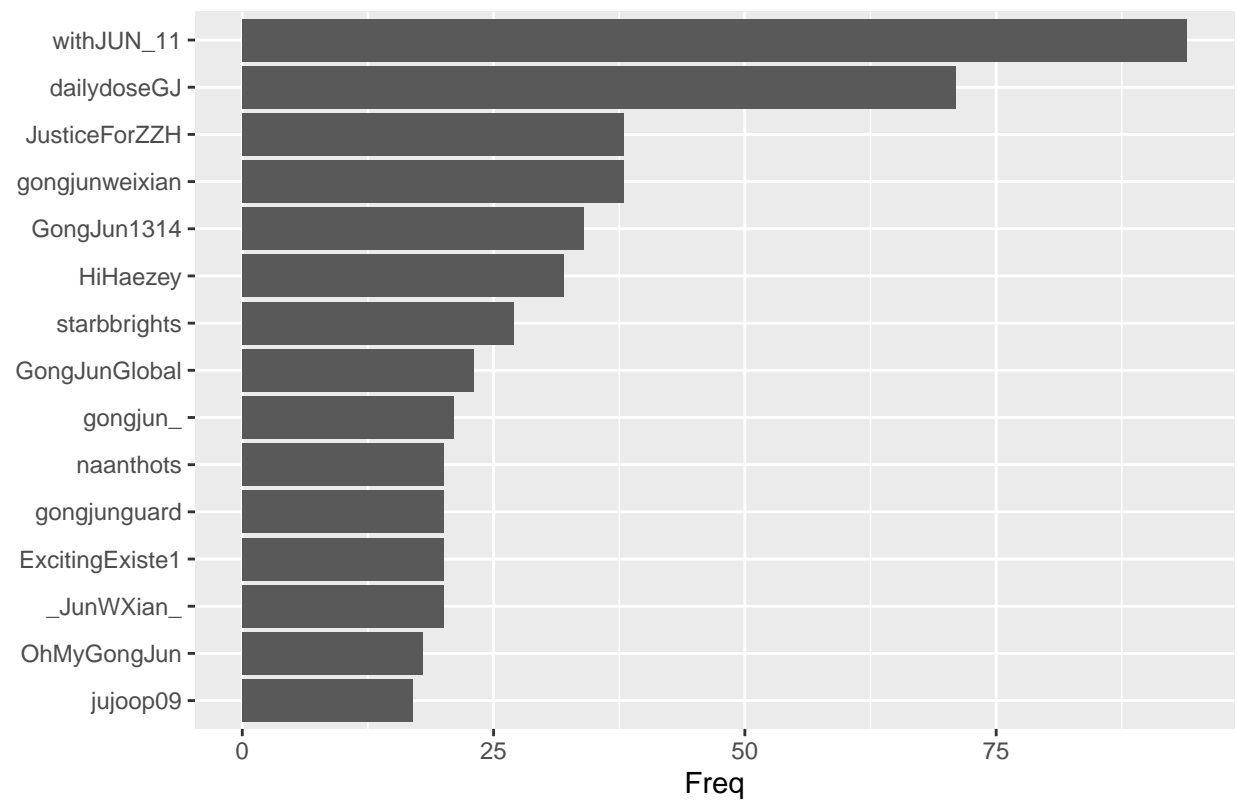
top accounts tweeting about #zhangzhehan , n= 1360 tweets



```
gj_tweets<-tweets_extract_plot(c("#gongjun"), "#gongjun")
```

```
## Selecting by Freq
```

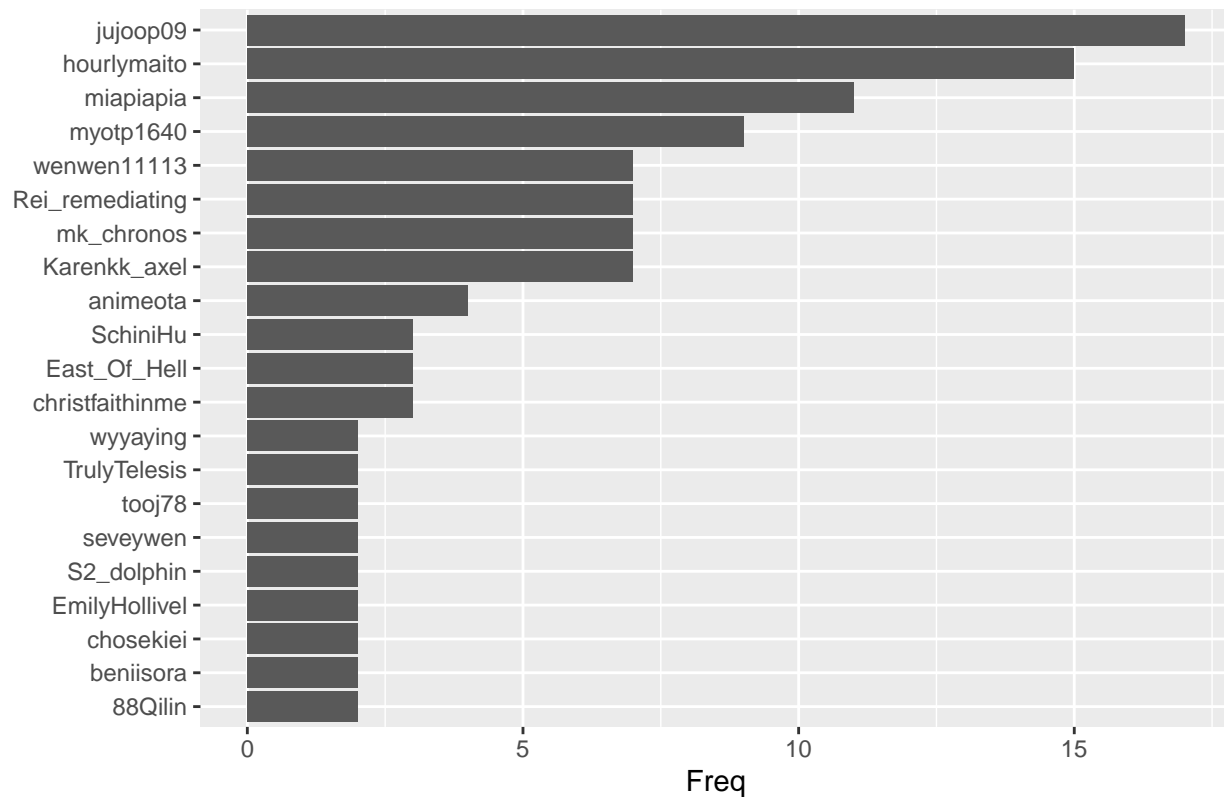
top accounts tweeting about #gongjun , n= 1069 tweets



```
jz_tweets<-tweets_extract_plot(c("#junzhe"), "#junzhe")
```

```
## Selecting by Freq
```

top accounts tweeting about #junzhe , n= 174 tweets



Quick info on which timespan the datasets cover and how many tweets per day

```
#calculate rate of tweets per hour
tweet_stats<-function(data,title){
  print(paste("dataset:",title))
  print(paste("number of tweets:",nrow(data)))
  start<-data[nrow(data),]$created_at
  end<-data[1,$created_at
  print(paste("start:",start))
  print(paste("end:",end))
  diff = end - start
  print(diff)
  #tweets per day
  tw_p_day<-round(nrow(data)/as.numeric(diff),2)
  print(paste("tweets per day:",tw_p_day))
  return(tw_p_day)
}

st_zzh<-tweet_stats(zzh_tweets,"#zhangzhehan")
```

```
## [1] "dataset: #zhangzhehan"
## [1] "number of tweets: 1360"
## [1] "start: 2022-05-17 23:46:18"
## [1] "end: 2022-05-26 06:50:58"
## Time difference of 8.294907 days
## [1] "tweets per day: 163.96"
```

```
st_gj<-tweet_stats(gj_tweets,"#gongjun")
```

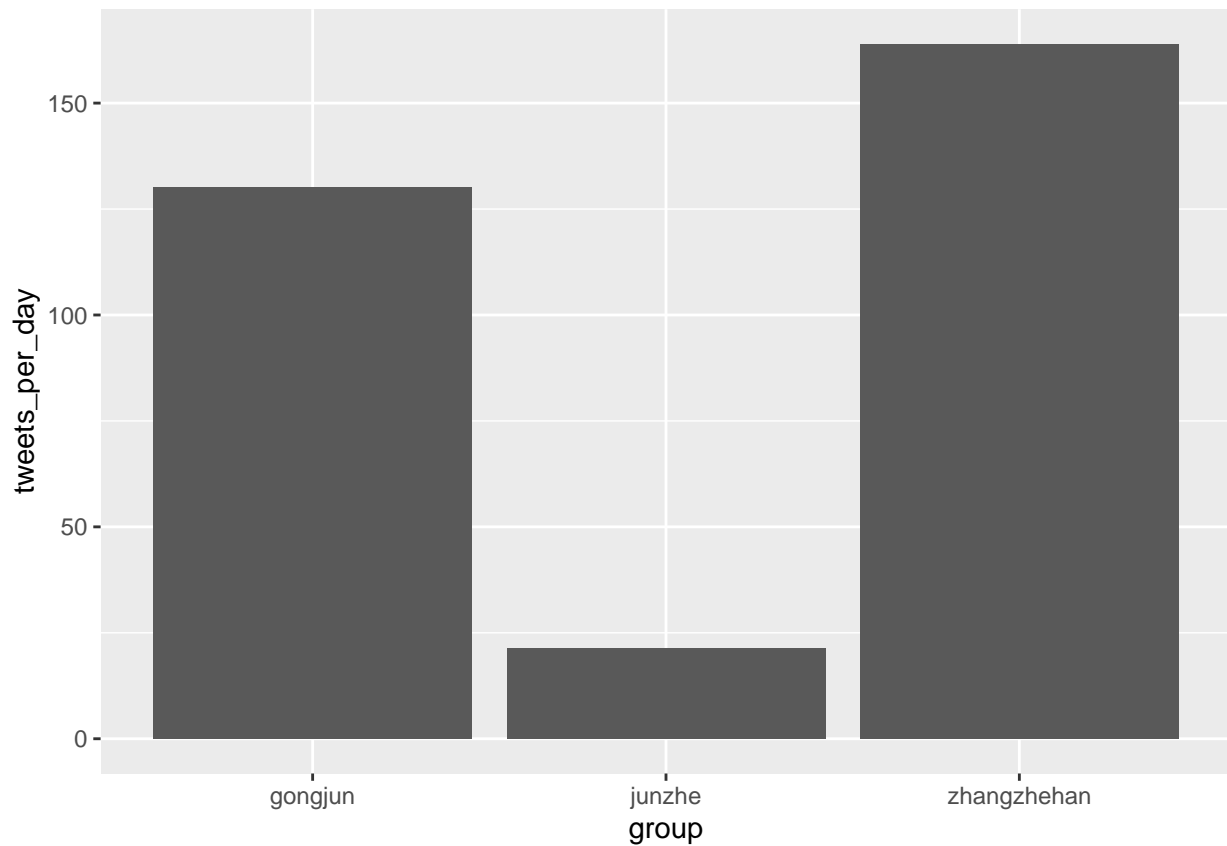
```
## [1] "dataset: #gongjun"  
## [1] "number of tweets: 1069"  
## [1] "start: 2022-05-18 01:39:45"  
## [1] "end: 2022-05-26 06:41:42"  
## Time difference of 8.209687 days  
## [1] "tweets per day: 130.21"
```

```
st_jz<-tweet_stats(jz_tweets,"#junzhe")
```

```
## [1] "dataset: #junzhe"  
## [1] "number of tweets: 174"  
## [1] "start: 2022-05-18 01:32:49"  
## [1] "end: 2022-05-26 06:00:47"  
## Time difference of 8.186088 days  
## [1] "tweets per day: 21.26"
```

```
bargraph<-data.frame(group=c("zhangzhehan","gongjun","junzhe"),tweets_per_day=c(st_zzh,st_gj,st_jz))
```

```
ggplot(bargraph,aes(x=group,y=tweets_per_day))+  
  geom_col()
```



Next let's make a Venn diagram to see the overlap between accounts tweeting about #junzh, #zhangzhehan or #gongjun. I have to use 2 Venn diagram packages because gplots gives me the numbers+intersections but only venneuler lets me make a beautiful diagram with proportional circle areas.

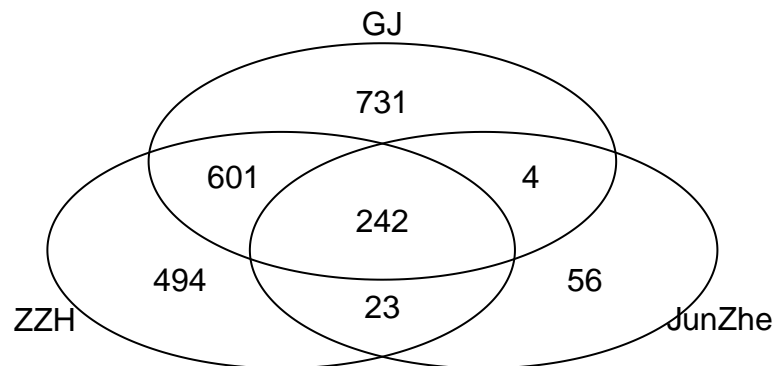
Note: This plot is not based on tweets containing the respective topics in combination. Instead it plots, e.g. how often accounts mentioning #zhangzhehan have also mentioned #gongjun, this can be in the same or different tweets.

```
###Venn diagram
library(gplots)
```

```
##
## Attaching package: 'gplots'

## The following object is masked from 'package:stats':
##
##      lowess
```

```
x <- list("ZZH" = zzh_tweets$screen_name,
         "JunZhe" = jz_tweets$screen_name,
         "GJ" = gj_tweets$screen_name)
v.table <- venn(x)
```



```

x <- attr(v.table,"intersections")
names_venn<-sapply(seq_along(x), function(i) paste(names(x)[[i]]))
numbers_venn<-lapply(seq_along(x), function(i) length(x[[i]]))
names_venn<-str_replace(names_venn, ":", "&")
names_venn<-str_replace(names_venn, ":", "&")
numbers_venn<-as.numeric(numbers_venn)
names(numbers_venn)<-names_venn

```

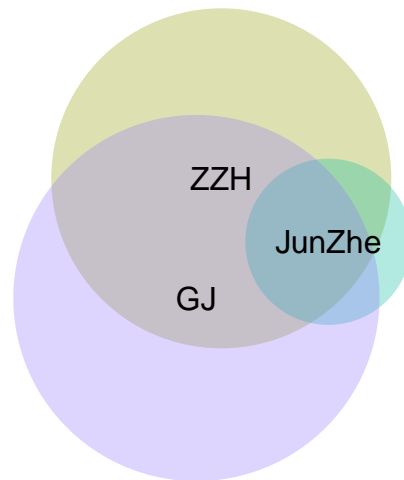
```
library(venneuler)
```

```
## Loading required package: rJava
```

```

v <- venneuler(numbers_venn)
plot(v)

```



Finally let's plot the top accounts tweeting about the respective topics+combinations. In these plots, for example the #zhangzhehan accounts are the accounts with the most tweets not mentioning the other topics.

```

###top accounts n venn diagram slices
topgraph<-function(index,data){
  n_tweets<-length(data)
  table(data)%>%
    as.data.frame() %>%
    arrange(desc(Freq)) %>%

```

```

top_n(15) %>%
mutate(data = reorder(data, Freq)) %>%
ggplot(aes(x = data, y = Freq)) +
geom_col() +
xlab(NULL) +
coord_flip() +
ggtitle(paste("top accounts tweeting about",index," n=",n_tweets,"tweets"))
}

```

```

x <- attr(v.table,"intersections")
lapply(seq_along(x), function(i) topgraph(names(x)[[i]], x[[i]]))

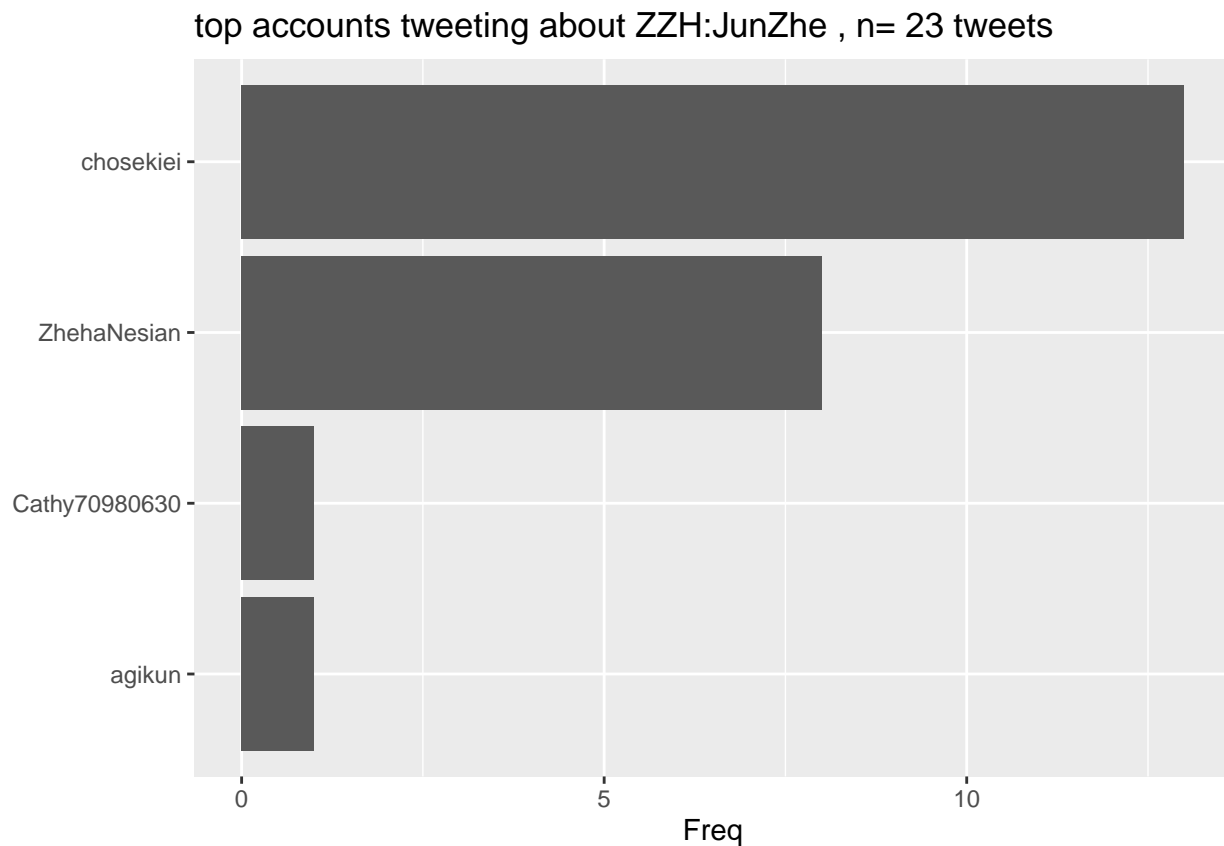
```

```

## Selecting by Freq
## Selecting by Freq
## Selecting by Freq
## Selecting by Freq
## Selecting by Freq
## Selecting by Freq
## Selecting by Freq

```

```
## [[1]]
```

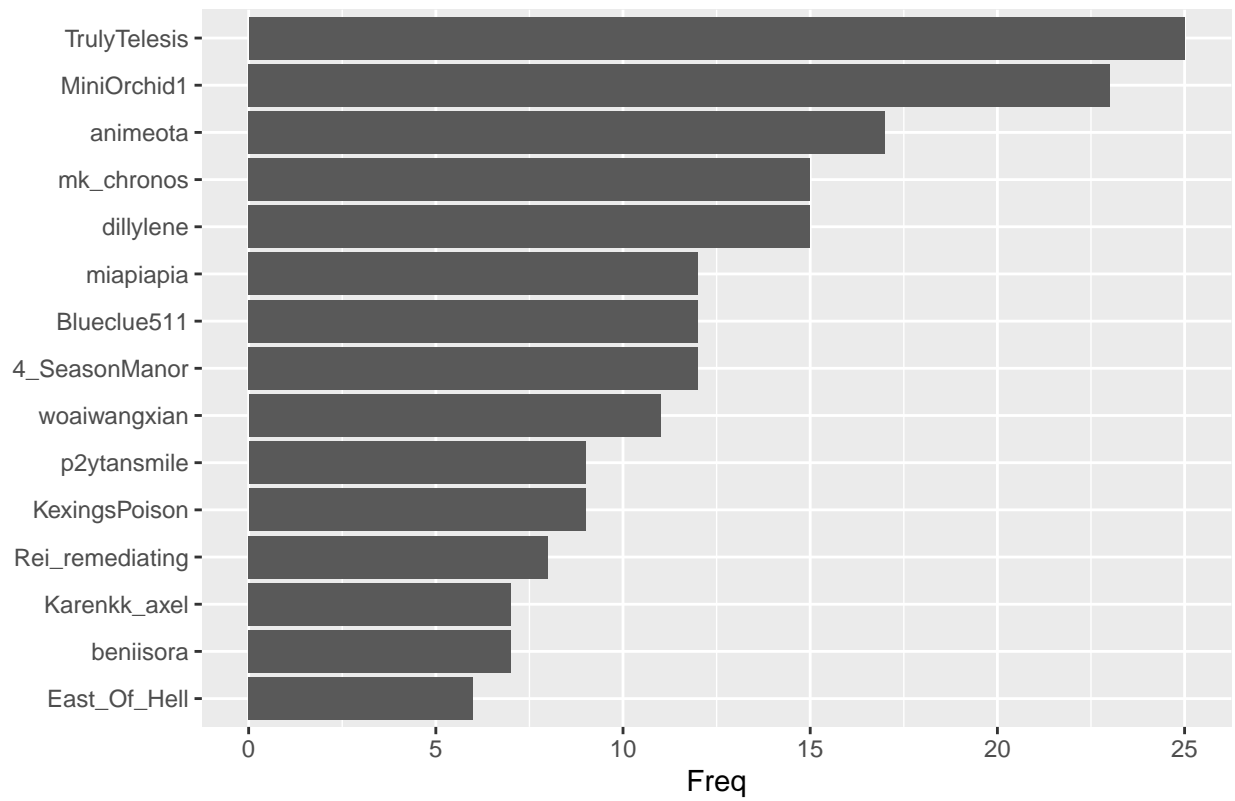


```

##
## [[2]]

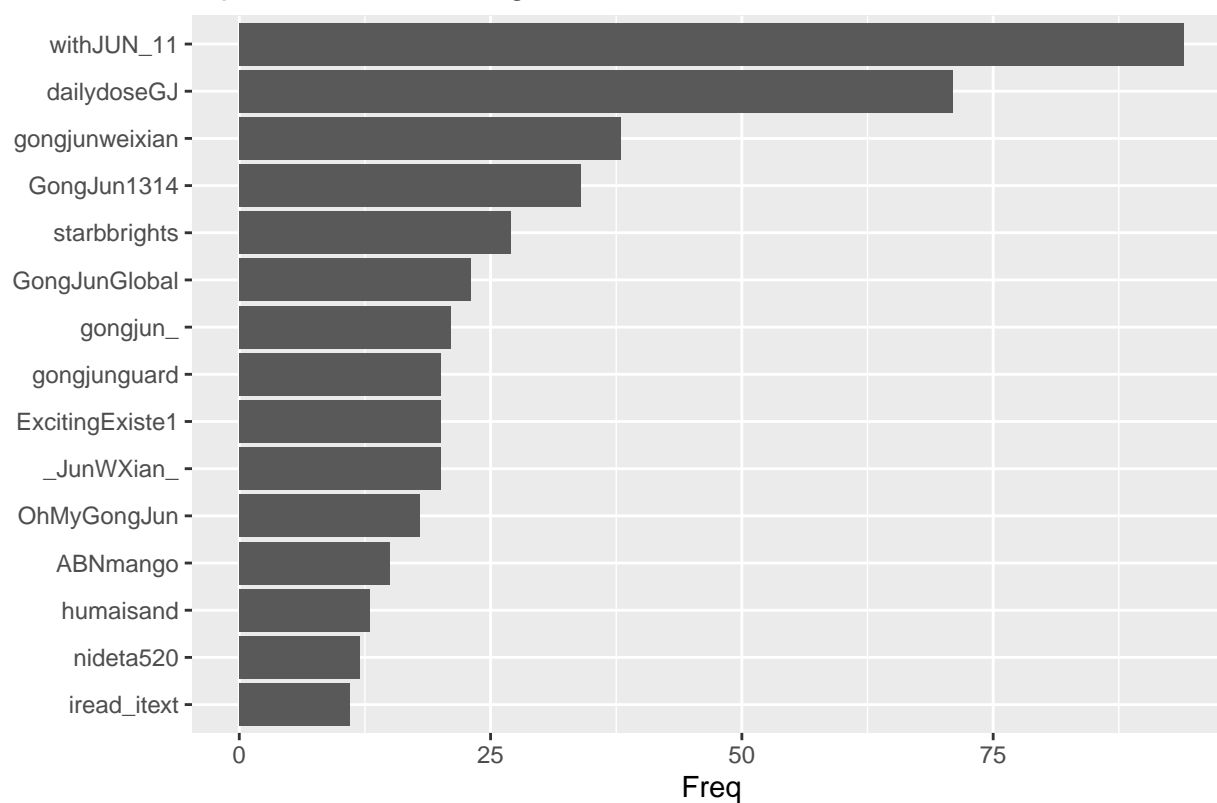
```


top accounts tweeting about ZZH:JunZhe:GJ , n= 242 tweets



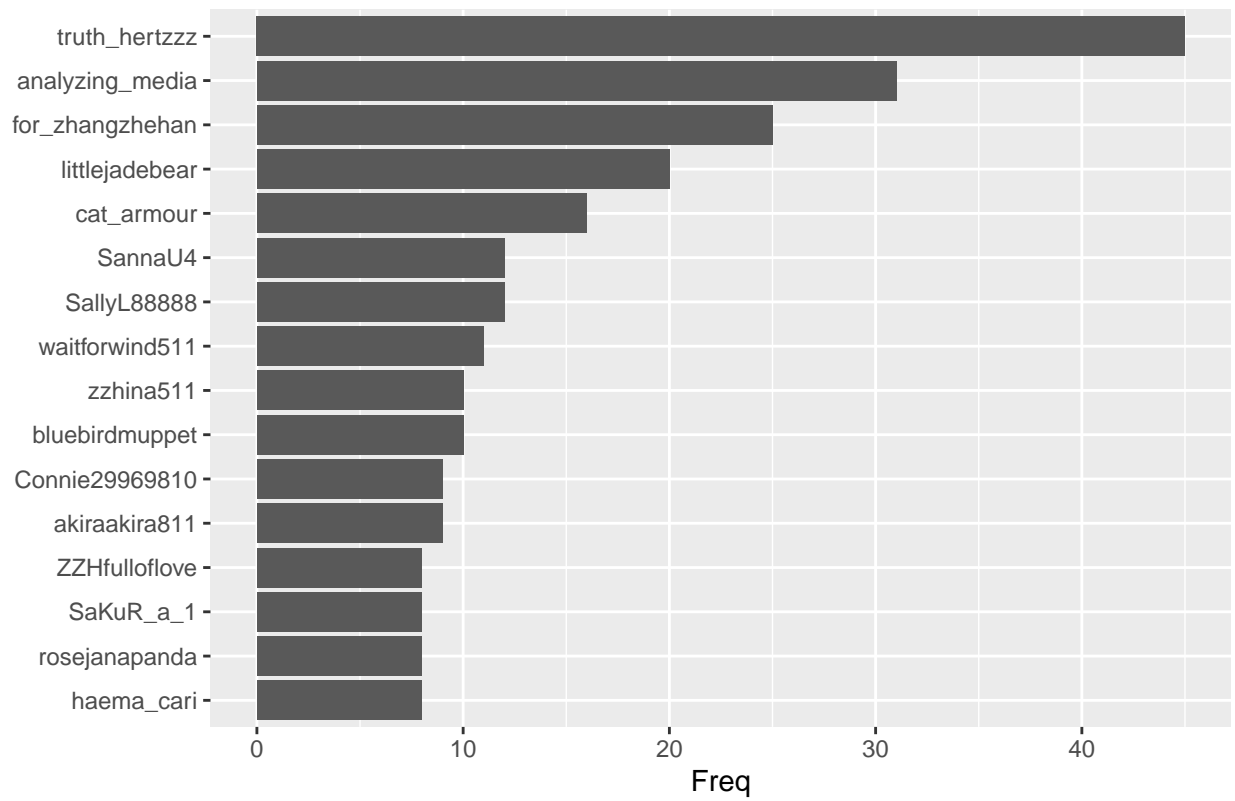
```
##  
## [[3]]
```

top accounts tweeting about GJ , n= 731 tweets



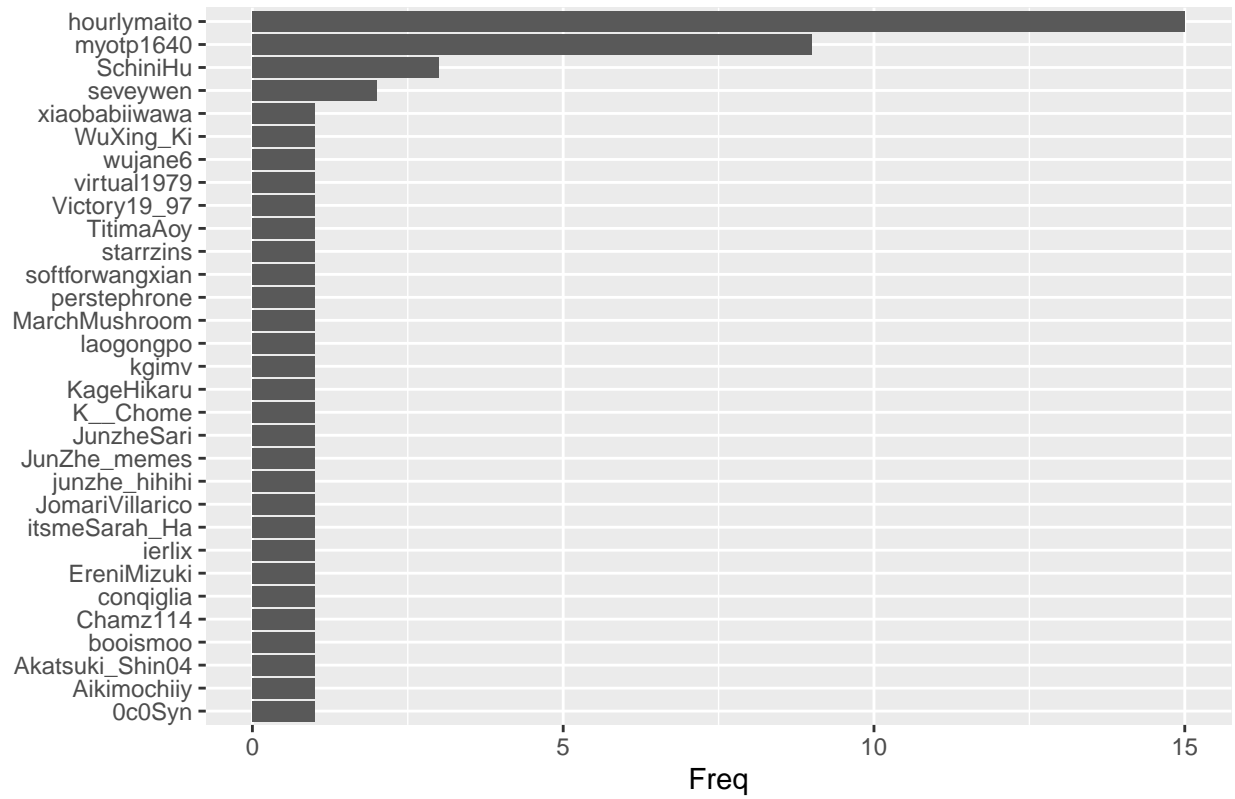
```
##
## [[4]]
```

top accounts tweeting about ZZH , n= 494 tweets



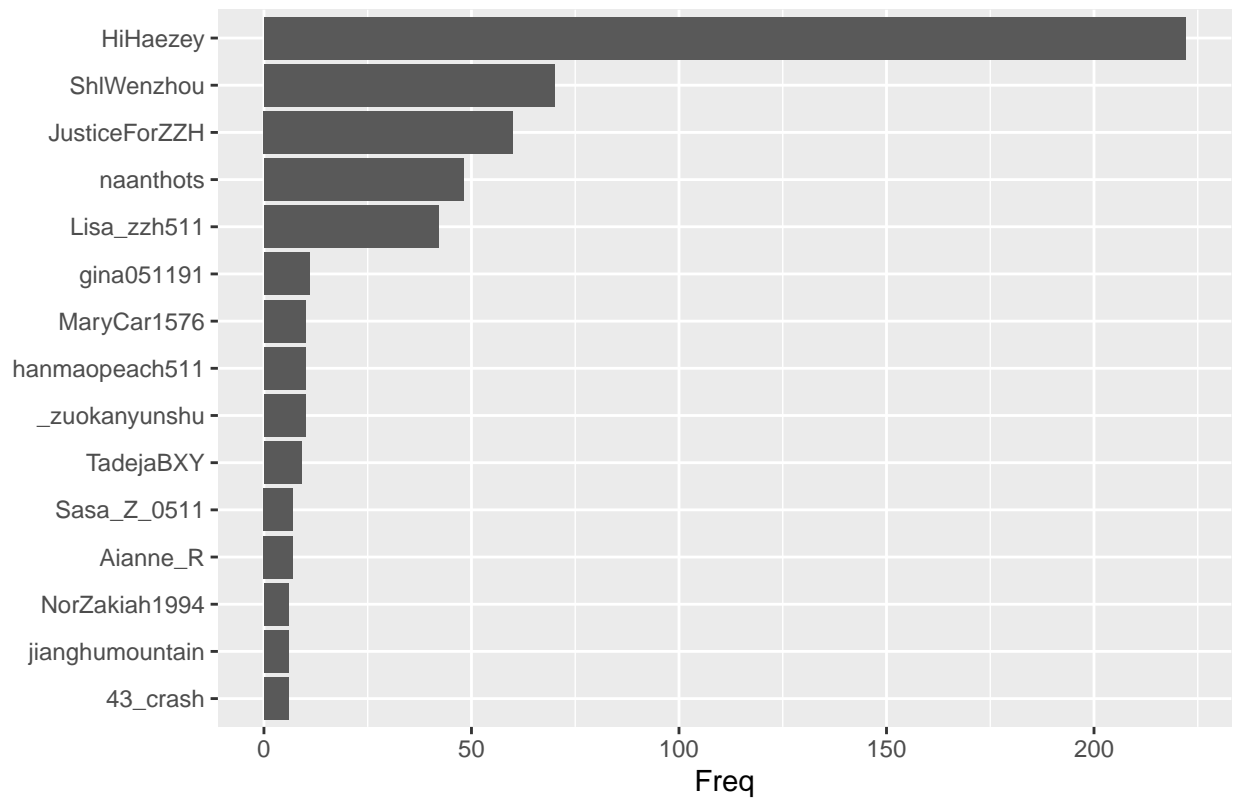
```
##  
## [[5]]
```

top accounts tweeting about JunZhe , n= 56 tweets



```
##
## [[6]]
```

top accounts tweeting about ZZH:GJ , n= 601 tweets



```
##  
## [[7]]
```

top accounts tweeting about JunZhe:GJ , n= 4 tweets

