Twitter 資料探勘：
環保少女與香港示威者討論熱度比較

第十六組 一七

黃顥	經濟一
陳宇澤	經濟一
葉秀軒	經濟一
蕭羽騏	會計一

2020.01.11

目錄

1. 簡介

1.1 主題構想與目標

當初構想主題時，備受爭議的瑞典環保少女——格蕾塔·桑伯格（Greta Thunberg），獲頒時代雜誌2019年度風雲人物。而在所有入圍名單中，香港示威者吸引了我們的目光，在2019年的下半年，香港反送中運動成為世界矚目的事件，根據時代雜誌統計，有超過30%的讀者將票投給了香港示威者，成為年度風雲人物讀者投票的大贏家。此消息一出，引起了許多網友的不滿，我們也對此產生了疑問，究竟兩者的討論度誰比較高? 因此我們決定用大型社群網站—— Twitter 資料探勘一探究竟，找出兩者在我們生活中的討論熱度及評價。

1.2 流程

主題構想與研究方向：
決定主題，訂出Twitter資料探勘研究方向，圖表分析 包括：文字雲，雙方討論熱度，被轉發次數分析圖表，

Twitter 資料取得：
申請Twitter API 取得資料，觀察資料取得對象，研究Twitter資料探勘限制

使用程式並觀察結果：
用R語言搜尋資料，並做出圖表，觀察結果與假設是否相符

提出結論：
利用圖表與資料，對比時事，提出結論並製造出報告與海報

2. 方法

2.1 資料取得與套件使用

在Twitter網站申請Twitter API，取得Twitter爬蟲許可，得到金鑰與密碼，並利用R開始取得資料。

文字雲運用 tm 以及 twitteR 套件， 其餘統計圖表則利用 rtweet 與 tidyverse 進行 資料的爬取與整理，加以ggplot2製圖。

2.2 原始碼運作說明

Twitter API認證 ：

```
#twitter API authorization process
consumer_key <- "NE7CD02PNt7nAxQ1cTVlFxRf7"
consumer_secret <-
"VpZwEaeN4cRRK64VxBZhKDCFZyVyOeyofOruZRqCM7BO0bs3ai"
access_token <- "1207590157643202560-
Tg2SdoeuD7CIg9LCYkVhNnqsFPhREX"
access_secret <-
"qvjmT7i4uWgIGKHZUfakRuWhiiE5zzDejKMhOprIE8BGx"
setup_twitter_oauth(consumer_key, consumer_secret,
access_token, access_secret)
```

文字雲程式碼(greta)：

```
require(twitteR)
require(RCurl)
require(tm)
require(wordcloud)

#search examples
gretathunberg_tweets <- searchTwitter("greta+thunberg",
lang="en", n=500, resultType = "recent")

#convert list to vector
gretathunberg_tweets_text <- sapply(gretathunberg_tweets,
function(x) x$getText())

#create corpus from vector od tweets
gretathunberg_corpus <-
Corpus(VectorSource(gretathunberg_tweets_text))
gretathunberg_corpus
inspect(gretathunberg_corpus[1])

#cleaning data
gretathunberg_clean <- tm_map(gretathunberg_corpus,
removePunctuation)
gretathunberg_clean <- tm_map(gretathunberg_clean,
content_transformer(tolower))
gretathunberg_clean <- tm_map(gretathunberg_clean,
removeWords, stopwords("english"))
gretathunberg_clean <- tm_map(gretathunberg_clean,
removeNumbers)
gretathunberg_clean <- tm_map(gretathunberg_clean,
stripWhitespace)
gretathunberg_clean <- tm_map(gretathunberg_clean,
removeWords,
c("greta","thunberg","gretathunberg","the","@gretathunberg"))

#create a wordcloud
wordcloud(gretathunberg_clean, random.order=F, scale =
c(3,0.5))
```

文字雲程式碼(hongkong)

```
#search examples #1000筆資料
hongkong_tweets <- searchTwitter("hongkong",
lang="en", n=1000, resultType = "recent")

#convert list to vector
hongkong_tweets_text <- sapply(hongkong_tweets,
function(x) x$getText())

#create corpus from vector od tweets
hongkong_corpus <-
Corpus(VectorSource(hongkong_tweets_text))
hongkong_corpus
inspect(hongkong_corpus[1])
#cleaning data
hongkong_clean <- tm_map(hongkong_corpus,
removePunctuation)
hongkong_clean <- tm_map(hongkong_clean,
content_transformer(tolower))
hongkong_clean <- tm_map(hongkong_clean, removeWords,
stopwords("english"))
hongkong_clean <- tm_map(hongkong_clean,
removeNumbers)
hongkong_clean <- tm_map(hongkong_clean,
stripWhitespace)
hongkong_clean <- tm_map(hongkong_clean, removeWords,
c("hongkong","the","hong","kong"))

#create a wordcloud
wordcloud(hongkong_clean, random.order=F, scale =
c(3,0.5))
```

討論熱度分析(hongkong) :

```
library(rtweet)
library(dplyr)
library(ggplot2)

#search tweets - HK
Hong_Kong <- search_tweets("HongKong",n=3200)

#select column
info <- Hong_Kong%>%
        select(created_at,retweet_count)%>%
        arrange(created_at)

#create chart
protest <- ggplot(info)+

geom_line(mapping=aes(x=created_at,y=retweet_count),co
lor=3)
```

討論熱度分析(greta):

```
#search tweets - greta
GretaThunberg <- search_tweets("GretaThunberg",n=3200)

#select column
info2 <- GretaThunberg%>%

select(created_at,favorite_count,retweet_count)%>%
         arrange(created_at)

#create chart
climate <- ggplot(info2)+

geom_line(mapping=aes(x=created_at,y=retweet_count),co
lor=3)
```

發文者分布地區分析(hongkong)：

```
install.packages("rtweet")
library(rtweet)
library(ggmap)
library(tidyverse)

#Google API
register_google(key =
"AIzaSyAOYjvvwkYxqM5lVSxDtdTAd_kNieQT964", write = TRUE)

#爬10000則有「＃hongkong」的貼文，找location
hongkong<- search_tweets("#hongkong",10000)
user_info <- lookup_users(unique(hongkong$user_id))
discard(user_info$location, `==`, "") %>%
  ggmap::geocode() -> coded
coded$location <- discard(user_info$location, `==`, "")
user_info <- left_join(user_info, coded, "location")

#篩選出locaiotn不為空白的值
tag_hkk_location <- user_info %>%
  select(location) %>%filter(location !="")

#計算「相同發訊位置」的貼文
tag_hkk_location <-
as.data.frame(table(tag_hkk_location$location))
#欄位「Var1」改成「Location」
colnames(tag_hkk_location)[1] <- "Location"

#按照tag次數由大到小排序
tag_hkk <- tag_hkk_location %>% arrange(desc(Freq))

#取前10
tag_hkk <- tag_hkk[1:10, 1:2]

#畫出圖形
ggplot(tag_hkk)+
  geom_col(mapping = aes(x = Location, y = Freq))
```

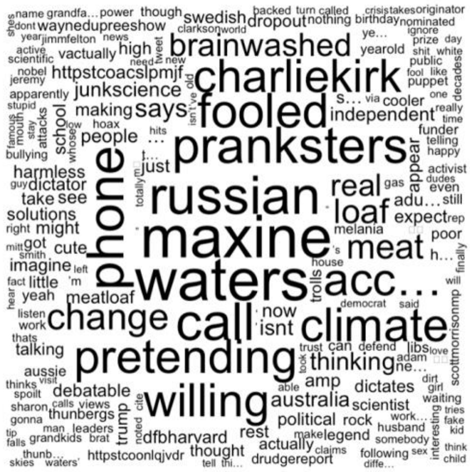
發文者分布地區分析(hongkong)：

```
#爬10000則有「#gretathunberg」的貼文，找location
greta <- search_tweets("#gretathunberg",10000)
user_info <- lookup_users(unique(greta$user_id))
discard(user_info$location, `==`, "") %>%
  ggmap::geocode() -> coded

coded$location <- discard(user_info$location, `==`, "")
user_info <- left_join(user_info, coded, "location")

#篩選出locaiotn不為空白的值
tag_greta_location <- user_info %>%
  select(location) %>%
  filter(location !="")

#計算「相同發訊位置」的貼文
tag_greta_location <-
as.data.frame(table(tag_greta_location$location))

#欄位「Var1」改成「Location」
colnames(tag_greta_location)[1] <- "Location"

#按照tag次數由大到小排序
tag_greta <- tag_greta_location %>%
  arrange(desc(Freq))

#取前10
tag_greta <- tag_greta[1:10, 1:2]
tag_greta

#作圖
ggplot(tag_greta)+
  geom_col(mapping = aes(x = Location, y = Freq))
```

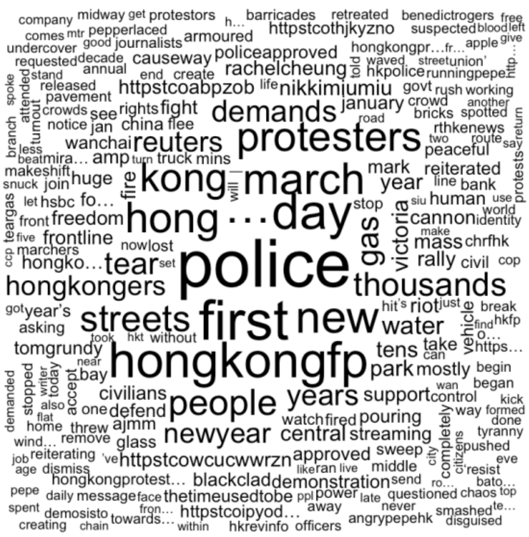
3. 結果

1. 文字雲

▲ 1/2 Greta Thunburg 文字雲



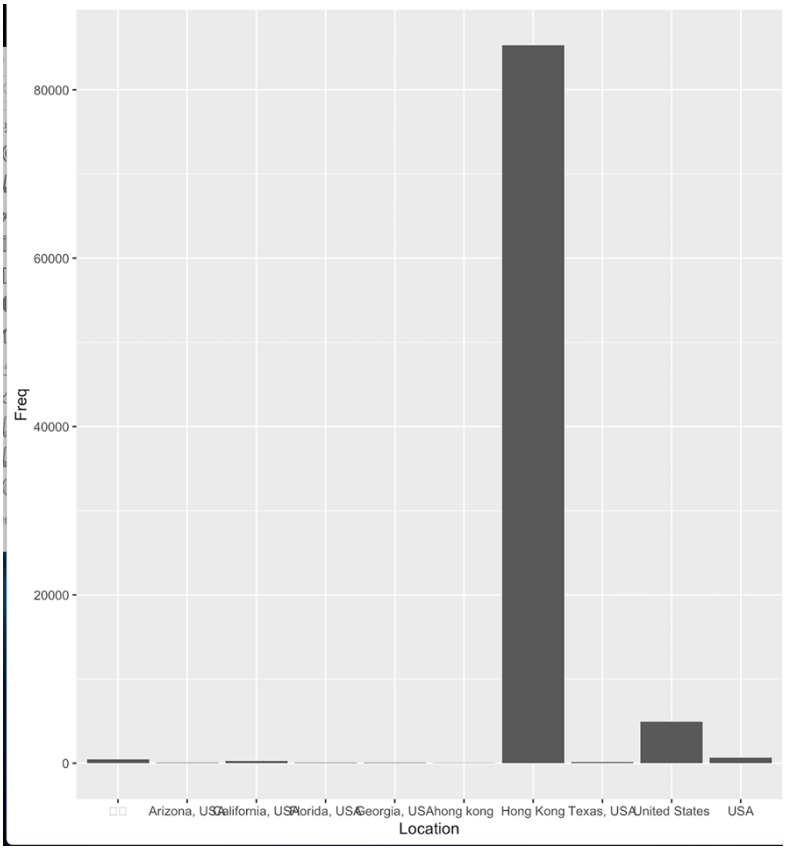
▲ 1/6 Greta Thunburg 文字雲



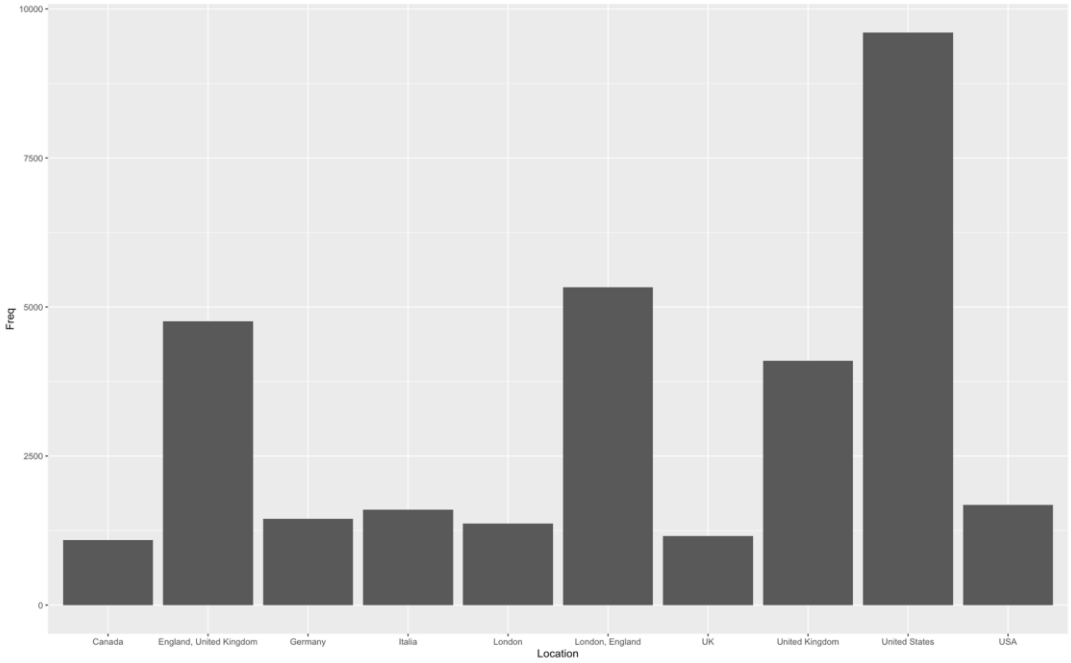
▲ 1/2 Honkong 文字雲

▲ 1/6 Hongkong 文字雲

2. 貼文發文者分布地區

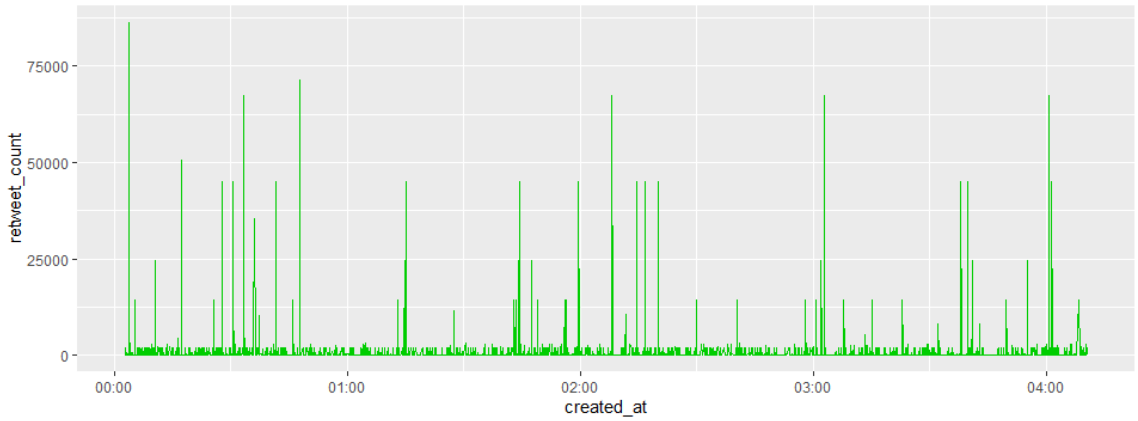


▲ Hongkong發文者分布地區圖表

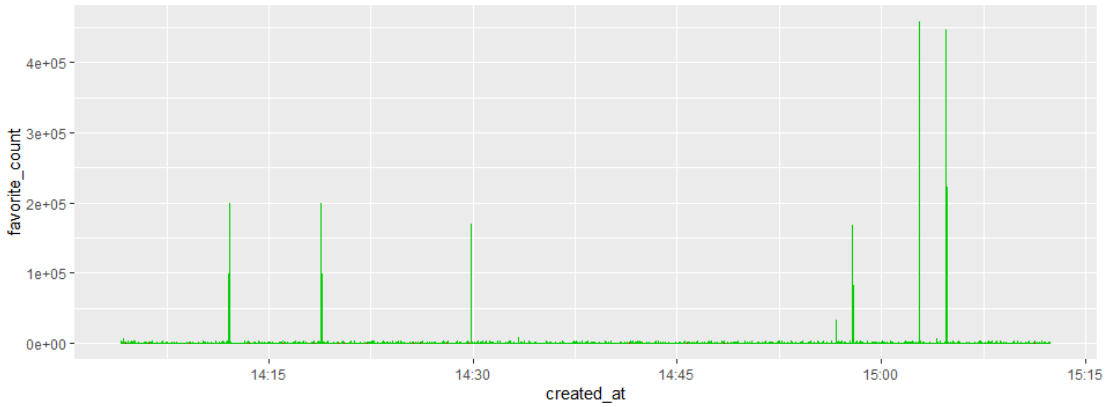


▲ Greta Thunburg發文者分布地區圖表

3. 熱度分析比較



▲ Greta Thunberg 熱度分析圖



▲ Hongkong 熱度分析圖

4. 討論與貢獻

4.1 改善重點

在我們的研究當中也存在著許多需要改善的地方，下面列出了幾點我們需要改進的地方：

1. 討論熱度中未考慮不同貼文的影響

我們在討論熱度的研究中，僅用貼文次數做熱度分析，並未分析各貼文討論流量帶來的影響以及該貼文的按讚數及留言並加以加權，這部分同時也需要更進一步的程式能力。

2. 程式能力與統計知識的侷限

我們原來要做影響力的分析，但因為我們的程式能力無法有效解決產生的問題，十分可惜。另外因為統計知識的侷限，我們無法進行進一步的研究，只能以表面上的情形加以研究。

3. Twitter使用區域及語言造成的樣本限制

在我們的研究中，我們以Twitter觀察全世界的熱度，但在爬取資料的過程中我們僅爬取了英文的文章，樣本數量並未能概括全世界，而Twitter使用者中以歐美地區為大多數，樣本數量較不足。

以上皆是我們可以改進且未來能努力的方向。

4.2 貢獻與發現

根據貼文區域性的長條圖，數據呈現hongkong 的數量明顯多於greta ，但可以發現多數「#hongkong」來自香港本地，相對於「#gretathunberg」，呈現較為區域化的現象，亞洲國家感受度相對強烈。另外，透過文字雲可以發現與桑伯格 相關的高頻字出現「brainwashed」、「fooled」 等等較為負面之詞彙；香港部分則出現意料 之中的「police」、「arrested」等字詞。最後， 再由折線圖可得知，隨著時間變化，Greta 的關注度是相對來說較穩定的，香港會因為特定的文突然增加關注度（特定的時間點或 發生特定事件），但是就每篇文被retweet的 總數來看，香港是大於greta的（以爬取資料當下的結果而言）。

4.3 結論

由於資料爬取上的限制（7-9天）以及每次爬取所獲得資料上的差異，我們僅能從有限的資料中得出部分結論：

在Twitter上，關於香港的發文數是大於Greta Thunburg 的，但從發文者所在地區來看，大多數還是香港人較關心，而頻率上來看則是Greta Thunberg 較高。雖然Twitter無法代表整個社會群體，但從Twitter中雖然Twitter無法代表整個社會群體，但從Twitter中還是可以發現，Greta有她一定程度的影響力（無論是正面抑或是反面），且助長了環保議題在世界各地的討論度（由長條圖可以略知一二）；即便作為一個略有爭議性的人物，時代雜誌的選擇也並非毫無緣由。...而資料分析上還有諸多當時未考慮進去的因素，例如正負面詞彙指稱的對象是誰（可能並非是指被標記的對象）等等，不完整的部分還請助教與教授多多指教。

5. 附錄

5.1 參考資料

https://www.earthdatascience.org/courses/earth-analytics/get-data-using-apis/use-twitter-api-r/?fbclid=IwAR0a2s3sa4Wzg3-e_aQG_vQ0eUGO7ZlhWOxaE2R5yF9-yd8VmPihQU-cVoE

https://rud.is/books/21-recipes/geocoding-locations-from-profiles-or-elsewhere.html?fbclid=IwAR0wjtzwjfwIZ9sua8Cfq8lslJkql5qDOszWwUHXtq8FrQsyS9bdB5gN2bI

https://rtweet-workshop.mikewk.com/?fbclid=IwAR0LpmWES1kv34uwCT_5YxE2dB2_shxWF-fBzBZTk_fZ-wR296VRfG6wls8#12

https://towardsdatascience.com/a-guide-to-mining-and-analysing-tweets-with-r-2f56818fdd16

5.2 組員分工

海報製作：蕭羽騏
報告製作：黃顥
程式： 蕭羽騏45% 陳宇澤 20% 葉秀軒20% 黃顥15%