特朗普父女推特解密

Table of Contents

美國新晉總統唐納德·特朗普（Donald Trump）以其極端言論在一眾政客裡獨領風騷。端Lab曾於2016年撰文分析特朗普與其競選對手希拉里·柯林頓（Hillary Clinton）面對媒體採訪時不同的言論風格，發現特朗普發言多用簡單句型，且善於用第二人稱敘事獲取觀眾共鳴。

除去媒體採訪，推特發言亦是特朗普競選的宣傳的另一愨道。因此本文選取唐納德·特朗普（下簡稱特朗普）及其女兒伊萬卡·特朗普（下簡稱伊萬卡）的推特數據作為樣本，利用文本信息挖掘方法（text mining methods），來分析他們在社交網絡上展示的話語特點。

# setwd('/Users/yuqiongli/Desktop/odd17/HKODD17-Trump')  
# Text Mining with R  
# (http://tidytextmining.com/) was  
# extensively referred to in this  
# project.  
  
# install.packages('tidytext')  
# install.packages('lubridate')  
# install.packages('tidyr')  
# install.packages('purrr')  
# install.packages('readr') # YQ  
# install.packages('qdapRegex') # YQ  
  
library(lubridate)  
library(ggplot2)  
library(dplyr)  
library(readr)  
library(stringr)  
library(tidytext)  
library(qdapRegex)  
library(tidyr)  
library(scales)  
library(purrr)  
library(broom)  
library(gridExtra)  
  
###### Check distribution of their tweets (by  
###### time / by device)  
  
trumpTweets <- read.csv("./data/realdonaldtrump.csv",   
 header = TRUE, sep = ",", stringsAsFactors = FALSE)  
ivankaTweets <- read.csv("./data/ivankatrump.csv",   
 header = TRUE, sep = ",", stringsAsFactors = FALSE)  
trumpTweets$created\_at <- format(strptime(trumpTweets$created\_at,   
 "%a %b %d %H:%M:%S %z %Y"), "%Y-%m-%d %H:%M:%S %z")  
ivankaTweets$created\_at <- format(strptime(ivankaTweets$created\_at,   
 "%a %b %d %H:%M:%S %z %Y"), "%Y-%m-%d %H:%M:%S %z")  
  
allTweets <- bind\_rows(trumpTweets %>% mutate(person = "Donald"),   
 ivankaTweets %>% mutate(person = "Ivanka")) %>%   
 mutate(timestamp = ymd\_hms(created\_at))  
# table(allTweets$source)

## 話癆特朗普

從發推文總數量來看，特朗普遠勝其女伊萬卡。2009到2016年，特朗普發布推文總數量三萬餘條，平均每天十條。而其女伊萬卡發文總量為一萬一千餘條，平均每天三條。二人皆疑似推特深度用戶。

# YQ - Calculate frequency of Tweets,  
# check sources  
nrow(allTweets[which(allTweets$person ==   
 "Donald"), ])

## [1] 30563

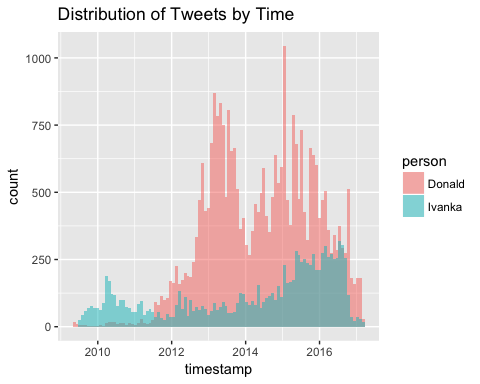
nrow(allTweets[which(allTweets$person ==   
 "Ivanka"), ])

## [1] 11263

allTweets$device <- ifelse(allTweets$source ==   
 "Twitter for Android", "Android", ifelse((allTweets$source ==   
 "Twitter for iPhone" | allTweets$source ==   
 "Twitter for iPad"), "iOS", ifelse(allTweets$source ==   
 "Twitter Web Client", "Web", "Otherwise")))  
  
allTweets$persondevice <- paste(allTweets$person,   
 allTweets$device, sep = "")

下圖顯示，特朗普推文数量從2011年起開始狂飆突進，並於2013年和2015年分別達到峰值。而伊萬卡的發文風格則更加穩健，只是在2016年時有所增加，可能是為特朗普競選造勢之故？

ggplot(allTweets, aes(x = timestamp, fill = person)) +   
 geom\_histogram(alpha = 0.5, position = "identity",   
 bins = 100) + ggtitle("Distribution of Tweets by Time")



## 特朗普家族最喜愛的推特發佈平台

特朗普與伊萬卡身為上流社會人士，當然不能滿足於用單一平台發布推文。

下表展示了特朗普父女較常使用的推特客戶端。他們在這些客戶端上發布的推文條數大於100。可以看出特朗普鍾愛安卓和網頁客戶端，而伊萬卡口味更加多元，除去推特網頁客戶端之外，更勇於嘗試 Buffer, Instagram, Sprout，BlackBerry，UberSocial 等各種較新的平台。她似乎並不鍾愛安卓手機。

# YQ - These two lines are clumsy  
# table(subset(trumpTweets$source,  
# table(trumpTweets$source)[trumpTweets$source]>100))  
trumpTweets$source %>% subset(., trumpTweets$source %>%   
 table(.)[.] > 100) %>% table(.)

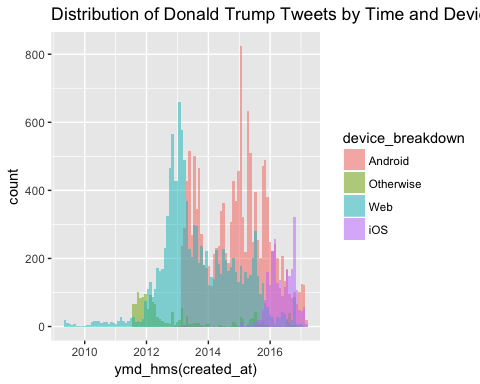
## .  
## Facebook Instagram TweetDeck   
## 105 133 483   
## TwitLonger Beta Twitter Web Client Twitter for Android   
## 405 12135 14520   
## Twitter for iPhone   
## 2485

ivankaTweets$source %>% subset(., ivankaTweets$source %>%   
 table(.)[.] > 100) %>% table(.)

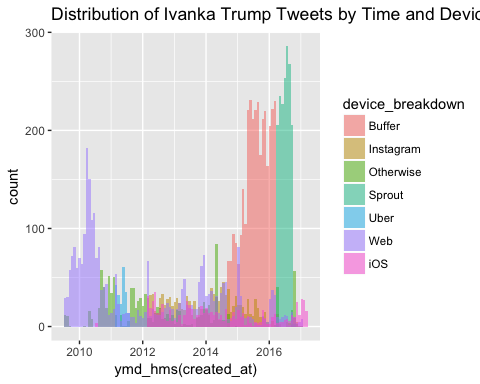
## .  
## Buffer Instagram   
## 3272 1188   
## Sprout Social Twitter Web Client   
## 1683 2998   
## Twitter for BlackBerry\302\256 Twitter for Websites   
## 441 108   
## Twitter for iPhone UberSocial Pro for iPhone   
## 811 207   
## \303\234berSocialOrig   
## 123

加入發文時間後，我們的分析發現，特朗普最初多使用网页推特发布消息，於2013年左右開始使用安卓手機系統客戶端發推特。2016左右，設備鳥槍換炮，用起了高大上的蘋果iOS系統客戶端。相比之下，伊萬卡發文的平台有較強階段性。2010年左右她與其父一樣是推特網頁版的忠實用戶，2011到2012年間還使用了其他設備和平台，如 Uber, BlackBerry等。2012年後她似乎開始喜愛上Instagram和iPhone平台。有意思的是，2015年她突然開始頻繁使用Buffer這款軟件，2016年則移情Sprout。這兩款均為社交網站管理平台，可以同時連結並管理Facebook、Twitter、Instagram等帳戶。從圖上來看，使用Buffer的時間段和Sprout的時間段並無交接，因此伊萬卡女士似乎更鍾意Sprout的用戶體驗。

## YQ - further break down of Trump tweets  
trumpTweets$device\_breakdown <- ifelse(trumpTweets$source ==   
 "Twitter for Android", "Android", ifelse((trumpTweets$source ==   
 "Twitter for iPhone" | trumpTweets$source ==   
 "Twitter for iPad"), "iOS", ifelse(trumpTweets$source ==   
 "Twitter Web Client", "Web", "Otherwise")))  
  
ggplot(trumpTweets, aes(x = ymd\_hms(created\_at),   
 fill = device\_breakdown)) + geom\_histogram(alpha = 0.5,   
 position = "identity", bins = 100) +   
 ggtitle("Distribution of Donald Trump Tweets by Time and Device")



# YQ - further breakdown the device types  
ivankaTweets$device\_breakdown <- ifelse(ivankaTweets$source ==   
 "Twitter for Android", "Android", ifelse((ivankaTweets$source ==   
 "Twitter for iPhone" | ivankaTweets$source ==   
 "Twitter for iPad"), "iOS", ifelse(ivankaTweets$source ==   
 "Twitter Web Client", "Web", ifelse(ivankaTweets$source ==   
 "Buffer", "Buffer", ifelse(ivankaTweets$source ==   
 "Instagram", "Instagram", ifelse(ivankaTweets$source ==   
 "Sprout Social", "Sprout", ifelse(ivankaTweets$source ==   
 "Twitter for BlackBerry<U+00AE>", "BlackBerry",   
 ifelse(ivankaTweets$source == "UberSocial Pro for iPhone" |   
 ivankaTweets$source == "<U+00DC>berSocialOrig",   
 "Uber", "Otherwise"))))))))  
  
ggplot(ivankaTweets, aes(x = ymd\_hms(created\_at),   
 fill = device\_breakdown)) + geom\_histogram(alpha = 0.5,   
 position = "identity", bins = 100) +   
 ggtitle("Distribution of Ivanka Trump Tweets by Time and Device")



## 詞彙偏好

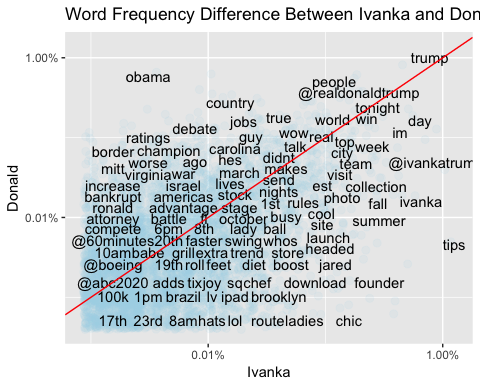
下圖列出不同詞彙在特朗普和伊萬卡的推特中分別出現的比例。可以發現，兩人共同姓氏“trump” ，以及一些常用詞"tonight","world","win", "forward"在雙方微博中出現的比例均較高。

同時，和上文發現類似的是，伊萬卡常用詞更偏向時尚、商業和生活，例如"advice","shops","chic","founder","intern" 等等。而特朗普有更多和政治、政策以及競選相關的詞彙，如"senator","endorsement","broken","complete"等。最能體現這一區別的一個例子是，伊萬卡大量使用了"tips"這一詞彙，儼然一位人生導師。而特朗普則大量使用了"Obama"，和他總統競選人的身份相符。

##### Clean allTweets  
  
allTweets$created\_at <- NULL  
  
# Keep only the completely recorded  
# tweets  
allTweets$incomplete <- grepl("\\(cont\\)",   
 allTweets$text)  
cleanedAllTweets <- allTweets[!allTweets$incomplete,   
 ]  
cleanedAllTweets$incomplete <- NULL  
  
# Keep only the tweets that are not  
# retweets >  
# table(cleanedAllTweets$is\_retweet)  
# False True 31437 9935  
cleanedAllTweets <- cleanedAllTweets[!as.logical(cleanedAllTweets$is\_retweet),   
 ]  
cleanedAllTweets$is\_retweet <- NULL  
  
# Extract URLs from tweets  
rm\_twitter\_n\_url <- rm\_(pattern = pastex("@rm\_twitter\_url",   
 "@rm\_url"))  
cleanedAllTweets$urls <- unlist(sapply(rm\_twitter\_n\_url(cleanedAllTweets$text,   
 extract = TRUE), function(x) return(paste(x,   
 collapse = "\t"))))  
cleanedAllTweets$text <- rm\_twitter\_n\_url(cleanedAllTweets$text)  
  
# Extract hashtags from tweets  
cleanedAllTweets$hashtags <- unlist(sapply(rm\_hash(cleanedAllTweets$text,   
 extract = TRUE), function(x) return(paste(x,   
 collapse = "\t"))))  
cleanedAllTweets$text <- rm\_hash(cleanedAllTweets$text)  
  
# Remove quotation marks and collapse  
# multiple whitespaces  
cleanedAllTweets$text <- gsub("'", "", cleanedAllTweets$text)  
cleanedAllTweets$text <- gsub("\"", "", cleanedAllTweets$text)  
cleanedAllTweets$text <- gsub("\\s+", " ",   
 str\_trim(cleanedAllTweets$text))  
  
### Tidy tweets using tidytext  
tidy\_tweets <- cleanedAllTweets %>% mutate(text = str\_replace\_all(text,   
 "https://t.co/[A-Za-z\\d]+|http://[A-Za-z\\d]+|&amp;|&lt;|&gt;|RT|https",   
 "")) %>% unnest\_tokens(word, text, token = "regex",   
 pattern = "([^A-Za-z\_\\d#@']|'(?![A-Za-z\_\\d#@]))") %>%   
 filter(!word %in% stop\_words$word, str\_detect(word,   
 "[a-z]"))  
  
##### Get word frequency using tidyr YQ-  
##### frequency here is n[i]/sum  
frequency <- tidy\_tweets %>% group\_by(person) %>%   
 count(word, sort = TRUE) %>% left\_join(tidy\_tweets %>%   
 group\_by(person) %>% summarise(total = n())) %>%   
 mutate(freq = n/total)  
# > frequency Source: local data frame  
# [29,589 x 5] Groups: person [2] person  
# word n total freq <chr> <chr> <int>  
# <int> <dbl> 1 Donald trump 1991 142627  
# 0.013959489 2 Donald obama 1128 142627  
# 0.007908741 3 Donald people 978 142627  
# 0.006857047 4 Ivanka tips 812 64851  
# 0.012521010 5 Donald time 715 142627  
# 0.005013076 6 Donald @realdonaldtrump  
# 706 142627 0.004949974 7 Donald donald  
# 673 142627 0.004718602 8 Donald america  
# 665 142627 0.004662511 9 Ivanka  
# @ivankatrump 569 64851 0.008773959 10  
# Donald president 559 142627 0.003919314  
# # ... with 29,579 more rows  
  
frequency <- frequency %>% select(person,   
 word, freq) %>% spread(person, freq) %>%   
 arrange(Donald, Ivanka)  
frequency

## # A tibble: 24,444 <U+00D7> 3  
## word Donald Ivanka  
## <chr> <dbl> <dbl>  
## 1 17th 7.011295e-06 1.541996e-05  
## 2 1k 7.011295e-06 1.541996e-05  
## 3 20k 7.011295e-06 1.541996e-05  
## 4 27th 7.011295e-06 1.541996e-05  
## 5 3x 7.011295e-06 1.541996e-05  
## 6 9am 7.011295e-06 1.541996e-05  
## 7 @10best 7.011295e-06 1.541996e-05  
## 8 @britneyspears 7.011295e-06 1.541996e-05  
## 9 @callmemrwayne 7.011295e-06 1.541996e-05  
## 10 @charlierose 7.011295e-06 1.541996e-05  
## # ... with 24,434 more rows

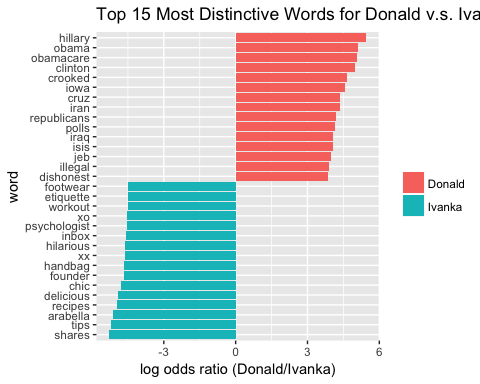
# # A tibble: 24,444 <U+00D7> 3 word  
# Donald Ivanka <chr> <dbl> <dbl> 1  
# @10best 7.011295e-06 1.541996e-05 2  
# @britneyspears 7.011295e-06  
# 1.541996e-05 3 @callmemrwayne  
# 7.011295e-06 1.541996e-05 4  
# @charlierose 7.011295e-06 1.541996e-05  
# 5 @claretourism 7.011295e-06  
# 1.541996e-05 6 @coachdanmullen  
# 7.011295e-06 1.541996e-05 7 @dabg3241  
# 7.011295e-06 1.541996e-05 8  
# @dailymailceleb 7.011295e-06  
# 1.541996e-05 9 @doral 7.011295e-06  
# 1.541996e-05 10 @drudge\_report  
# 7.011295e-06 1.541996e-05 # ... with  
# 24,434 more rows  
  
ggplot(frequency, aes(Ivanka, Donald)) +   
 geom\_jitter(alpha = 0.1, size = 2.5,   
 width = 0.25, height = 0.25, color = "lightblue") +   
 geom\_text(aes(label = word), check\_overlap = TRUE,   
 vjust = 1.5) + scale\_x\_log10(labels = percent\_format()) +   
 scale\_y\_log10(labels = percent\_format()) +   
 geom\_abline(color = "red") + ggtitle("Word Frequency Difference Between Ivanka and Donald Trump")



下圖顯示基於詞頻比對的特徵詞結果（注2），我們選出三十個雙方使用頻率相差最大的詞彙。紅色部分代表特朗普較常使用而伊萬卡不常使用的詞彙，藍色部分代表伊萬卡常用而其父不常用的詞彙。

由表看出，和上文結論類似，兩人最大區別在於特朗普經常提起競選相關詞彙，如"hillary","obama","obamacare","clinton"等。而伊萬卡常常提起时尚或生活相關詞彙，如"tips","shares","handbag"等。兩張詞彙雲圖則更直觀地反映出二者用詞的差異。這可能和二者政治家和時尚商人的不同身分有關。伊萬卡於2007年創立自己的珠寶時尚品牌。

##### Word usage  
word\_ratios <- tidy\_tweets %>% filter(!str\_detect(word,   
 "^@")) %>% count(word, person) %>% filter(sum(n) >=   
 10) %>% spread(person, n, fill = 0) %>%   
 ungroup() %>% mutate\_each(funs((. + 1)/sum(. +   
 1)), -word) %>% mutate(logratio = log(Donald/Ivanka)) %>%   
 arrange(desc(logratio))  
  
word\_ratios %>% group\_by(logratio < 0) %>%   
 top\_n(15, abs(logratio)) %>% ungroup() %>%   
 mutate(word = reorder(word, logratio)) %>%   
 ggplot(aes(word, logratio, fill = logratio <   
 0)) + geom\_col() + coord\_flip() +   
 ylab("log odds ratio (Donald/Ivanka)") +   
 scale\_fill\_discrete(name = "", labels = c("Donald",   
 "Ivanka")) + ggtitle("Top 15 Most Distinctive Words for Donald v.s. Ivanka Trump")



## 詞彙雲

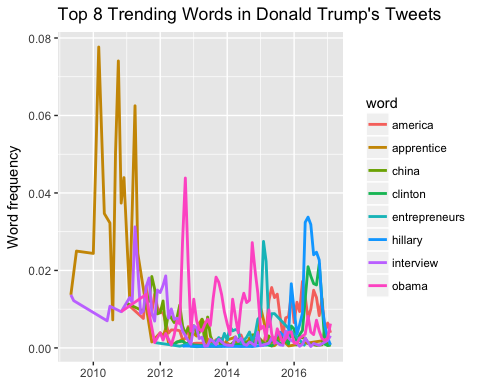
This part takes a long time to run so the word clouds are separately attached.

## Word cloud - Author: Yuqiong Li &  
## Yiming Li This part takes a long time  
## to run so the wordcloud is separately  
## attached.  
  
# install.packages('tm')  
# install.packages('wordcloud')  
# install.packages('RColorBrewer')  
# install.packages('slam')  
  
library(tm)  
library(wordcloud)  
library(RColorBrewer)  
  
# makeCloud <- function(docs, graphfile =  
# 'wordcloud.pdf') { # Convert the text  
# to lower case docs <- tm\_map(docs,  
# content\_transformer(tolower)) # Remove  
# numbers docs <- tm\_map(docs,  
# removeNumbers) # Remove english common  
# stopwords docs <- tm\_map(docs,  
# removeWords, stopwords('english')) #  
# Remove your own stop word # specify  
# your stopwords as a character vector  
# docs <- tm\_map(docs, removeWords,  
# c('the', 'get')) # docs <- tm\_map(docs,  
# content\_transformer(gsub), pattern =  
# 'thanks', replacement = 'thank',  
# fixed=TRUE) # Remove punctuations docs  
# <- tm\_map(docs, removePunctuation) #  
# Eliminate extra white spaces docs <-  
# tm\_map(docs, stripWhitespace) # Text  
# stemming docs <- tm\_map(docs,  
# stemDocument) dtm <-  
# TermDocumentMatrix(docs) m <-  
# as.matrix(dtm) v <-  
# sort(rowSums(m),decreasing=TRUE) d <-  
# data.frame(word = names(v),freq=v) #  
# head(d, 10) pdf(file = graphfile)  
# set.seed(1234) wordcloud(words =  
# d$word, freq = d$freq, min.freq = 1,  
# max.words=200, random.order=FALSE,  
# rot.per=0.35, colors=brewer.pal(8,  
# 'Dark2')) dev.off() return(list(docs =  
# docs, dtm = dtm, d = d)) }  
# load('tweets2.RData') trumpCorpus <-  
# makeCloud(Corpus(VectorSource(trumpTweetsV$cleaned)),  
# graphfile = 'trumpcloud.pdf')  
# ivankaCorpus <-  
# makeCloud(Corpus(VectorSource(ivankaTweetsV$cleaned)),  
# graphfile = 'ivankacloud.pdf')  
# save(file = 'tweets3.RData', list =  
# c('trumpCorpus', 'ivankaCorpus'))

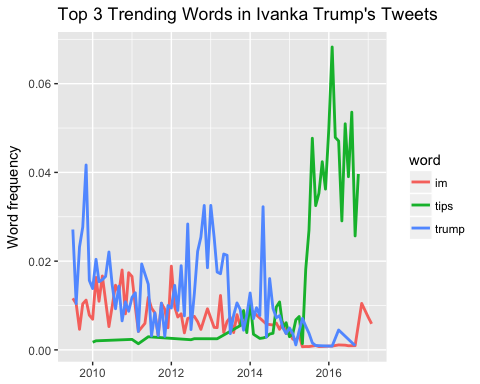
## 父女口癖變遷史

Trending words

words\_by\_time <- tidy\_tweets %>% filter(!str\_detect(word,   
 "^@")) %>% mutate(time\_floor = floor\_date(timestamp,   
 unit = "1 month")) %>% count(time\_floor,   
 person, word) %>% ungroup() %>% group\_by(person,   
 time\_floor) %>% mutate(time\_total = sum(n)) %>%   
 group\_by(word) %>% mutate(word\_total = sum(n)) %>%   
 ungroup() %>% rename(count = n) %>% filter(word\_total >   
 30)  
nested\_data <- words\_by\_time %>% nest(-word,   
 -person)  
  
nested\_models <- nested\_data %>% mutate(models = map(data,   
 ~glm(cbind(count, time\_total) ~ time\_floor,   
 ., family = "binomial")))  
  
slopes <- nested\_models %>% unnest(map(models,   
 tidy)) %>% filter(term == "time\_floor") %>%   
 mutate(adjusted.p.value = p.adjust(p.value))  
  
top\_slopes\_d <- slopes %>% filter(adjusted.p.value <   
 1e-20)  
words\_by\_time %>% inner\_join(top\_slopes\_d,   
 by = c("word", "person")) %>% filter(person ==   
 "Donald") %>% ggplot(aes(time\_floor,   
 count/time\_total, color = word)) + geom\_line(size = 1) +   
 labs(x = NULL, y = "Word frequency") +   
 ggtitle("Top 8 Trending Words in Donald Trump's Tweets")



top\_slopes\_i <- slopes %>% filter(adjusted.p.value <   
 1e-20)  
words\_by\_time %>% inner\_join(top\_slopes\_i,   
 by = c("word", "person")) %>% filter(person ==   
 "Ivanka") %>% ggplot(aes(time\_floor,   
 count/time\_total, color = word)) + geom\_line(size = 1) +   
 labs(x = NULL, y = "Word frequency") +   
 ggtitle("Top 3 Trending Words in Ivanka Trump's Tweets")



## 最易引發“轉發”和“喜歡”的詞彙

“轉發”和“喜歡”均為發布者和關注者的某種互動形式。因此，通過分析獲得更多“轉發”和“喜歡”的推特詞彙特點，既可以分析出發布者的心態，也可以分析出關注群體對特定內容的偏好。紐約時報2011年一項針對社交媒體的調查發現，用戶在社交媒體上分享信息主要有五種情況：分享娛樂性消息、自我包裝和認同、增強社交關係、進行對話，以及推廣新聞、產品信息等【注2】。

下圖展示了在特朗普和伊萬卡的推特數據裡，哪些詞彙容易引發更多轉發。特朗普的推特詞彙中，獲取轉發數最高的為"hamilton","praying"和"wikileaks"。而伊萬卡的則為"policy","family","theodore"。這些詞彙個人性較弱，更類似對特定消息的推廣，符合二人公眾人物的身分。

"Hamilton"的高出鏡率可能因為如下這條推特 -

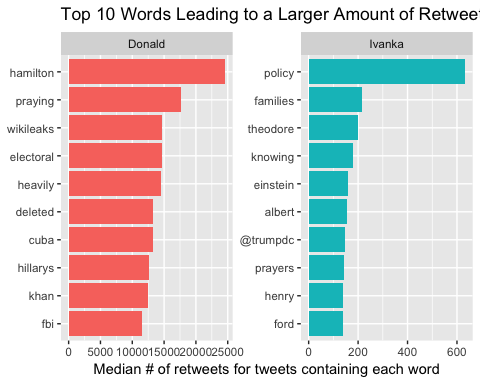
“Donald J. Trump ✔@realDonaldTrump The Theater must always be a safe and special place.The cast of Hamilton was very rude last night to a very good man, Mike Pence. Apologize! 9:56 PM - 19 Nov 2016 43,592 43,592 Retweets 149,734 149,734 likes”

在這條發布與2016年11月19日，獲得四萬三千多條轉發的推文中，特朗普指責音樂劇 “Hamilton”的劇組對副總統 Mike Pence 進行騷擾。Mike Pence 之前觀賞了該音樂劇。表演結束後，劇組成員當眾向他表達了對新當選政府的不信任，和對未來的期待。這一舉動似乎激怒了特朗普，他在推特上發文要求劇組向副總統道歉。

詞彙"praying"的信息量較少。"wikileaks"可能與特朗普競選時多次強調該網站揭露的希拉里郵件醜聞有關。可以想像，含有類似內容的推特因為具有新聞性，較易引發交流，從而引起轉發和討論。

關於伊萬卡的推特分析，“theodore”是她新出生兒子的名字，這也和“family”獲得較高轉發量相吻合。另一方面，這可能反映出美國文化對家庭價值的重視。與上文類似，包含詞彙“policy”的推特，可能因為其新聞性和內容性而獲得較高轉發。

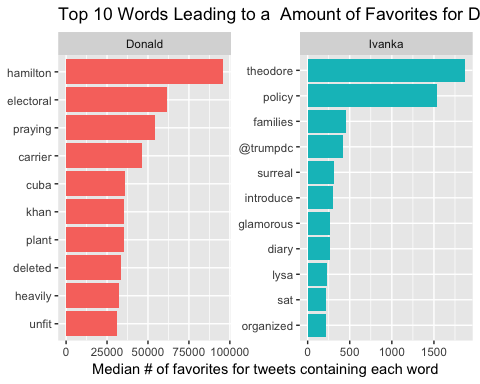
totals <- tidy\_tweets %>% group\_by(person,   
 id\_str) %>% summarise(rts = sum(retweet\_count)) %>%   
 group\_by(person) %>% summarise(total\_rts = sum(rts))  
# person total\_rts <chr> <int> 1 Donald  
# 270962155 2 Ivanka 3063530  
  
word\_by\_rts <- tidy\_tweets %>% group\_by(id\_str,   
 word, person) %>% summarise(rts = first(retweet\_count)) %>%   
 group\_by(person, word) %>% summarise(retweet\_count = median(rts),   
 uses = n()) %>% left\_join(totals) %>%   
 filter(retweet\_count != 0) %>% ungroup()  
  
# word\_by\_rts %>% filter(uses >= 5) %>%  
# arrange(desc(retweet\_count))  
  
word\_by\_rts %>% filter(uses >= 5) %>% group\_by(person) %>%   
 top\_n(10, retweet\_count) %>% arrange(retweet\_count) %>%   
 mutate(word = factor(word, unique(word))) %>%   
 ungroup() %>% ggplot(aes(word, retweet\_count,   
 fill = person)) + geom\_col(show.legend = FALSE) +   
 facet\_wrap(~person, scales = "free",   
 ncol = 2) + coord\_flip() + labs(x = NULL,   
 y = "Median # of retweets for tweets containing each word") +   
 ggtitle("Top 10 Words Leading to a Larger Amount of Retweets for Donald and Ivanka Trump")



## 最易引發“喜歡”的詞彙

關於社交媒體上用戶“喜歡”某條推特，或者“點贊”的行為，來自社交媒體管理網站Buffer（伊萬卡之前最愛）分析認為，點贊的行為動機可分為四種：類似線下交流時點頭等打招呼的行為，對自己認同的某些價值再度肯定，表達同情，以及獲取現實回報（如餐廳打折等）。分析特朗普和伊萬卡獲取“點贊”數最多的推特詞彙可以發現，兩人的前三名大致與獲取轉發數較高的推特相同。唯一區別在於特朗普發布的“electoral”詞彙亦獲得較多喜歡。這可能是支持者表達鼓勵的行為反映。

##### Favorites  
  
totals <- tidy\_tweets %>% group\_by(person,   
 id\_str) %>% summarise(favs = sum(favorite\_count)) %>%   
 group\_by(person) %>% summarise(total\_favs = sum(favs))  
# person total\_favs <chr> <int> 1 Donald  
# 704259573 2 Ivanka 11179329  
  
word\_by\_favs <- tidy\_tweets %>% group\_by(id\_str,   
 word, person) %>% summarise(favs = first(favorite\_count)) %>%   
 group\_by(person, word) %>% summarise(favorite\_count = median(favs),   
 uses = n()) %>% left\_join(totals) %>%   
 filter(favorite\_count != 0) %>% ungroup()  
  
# word\_by\_favs %>% filter(uses >= 5) %>%  
# arrange(desc(favorite\_count))  
  
word\_by\_favs %>% filter(uses >= 5) %>% group\_by(person) %>%   
 top\_n(10, favorite\_count) %>% arrange(favorite\_count) %>%   
 mutate(word = factor(word, unique(word))) %>%   
 ungroup() %>% ggplot(aes(word, favorite\_count,   
 fill = person)) + geom\_col(show.legend = FALSE) +   
 facet\_wrap(~person, scales = "free",   
 ncol = 2) + coord\_flip() + labs(x = NULL,   
 y = "Median # of favorites for tweets containing each word") +   
 ggtitle("Top 10 Words Leading to a Amount of Favorites for Donald and Ivanka Trump")



註釋：本文使用主要文本分析工具是R，部分分析方法參考網站 <http://tidytextmining.com/> 。

注【2】：<http://text-ex-machina.co.uk/blog/new-york-times-study.html>

注【3】：<https://blog.bufferapp.com/psychology-of-facebook>