105 學年第 2 學期
巨量資料分析技術與應用 課程

Term Project Report

第 8 組

0556080 黃昱銓

0556176 紀閎全

0556562 陳鴻君

0556555 林哲宇

一、 組員分工與各自執行細項

(Team members & Task allocation)

分工事項/組員	黃昱銓	紀閎全	陳鴻君	林哲宇
資料爬蟲	20%	10%	10%	60%
字串抽取	10%	70%	10%	10%
字頻分析	60%	20%	10%	10%
關聯性運算	10%	10%	70%	10%
網頁呈現	20%	10%	10%	60%
報告呈現	25%	25%	25%	25%

Note: 概念都是一起討論，占比10%為討論參與。

二、 計畫目標的問題

(Target problem)

現在是網路發達的世代，網路論壇對於事件或是人物都有歧異性極大的討論，我們期望透過爬取美國最大論壇reddit之討論串資訊得知自己喜歡的NBA球星們或隊伍在國外的評價（風向）為何。

並希望根據大量資料建立的文字模型，能夠透過同類球員的屬性（球風、位置）給予未知球員的評價預測。

三、 選用的資料集描述與觀察

(Descriptions of selected datasets, including the characteristics in terms of Big Data)

我們的資料來源為Reddit論壇之NBA討論版，平常聚集大量美國網民，根據正常的觀察下，我們發現網友的評論經常是沒有主詞、針對影像片段或是根據戰術做評論，在這樣的情況下，我們必須注意資料的處理。

在BigData領域內，爬蟲爬下來的資料是未經分類的大量文字及回應串，首先符合Volume（量）。並且每位網（ㄙㄨㄤ）民的意見並不一定相同，網路上的評論很常出現兩極化的現象，持續不斷的收集這些資料來精進我們的模型，以達到Velocity（實時），但都是以文字進行分析及評價，資料格式較無歧異度，雖常有連結及Emoji亂碼穿插，但並未完全符合Variety（格式）。

四、 針對問題設計的分析流程

(Analysis workflow)

a. 爬取Reddit資料（Python）:

利用PRAW(Python Reddit Api Wrapper)，我們可以輕易獲取reddit上任何一個subreddit之文章，但是因為該論壇API的規章，每秒獲取以及單一帳號獲取的量有所限制，我們申請多個帳號以及設定Crawler Bot不同之User Agent，利用分散式的運算來獲取足夠資料。

b. Comment特徵擷取（Python）:

利用NLTK(Natural Language Toolkit)的remove_stopword，除去Comment中英文語句常出現的字詞，並將剩餘的字詞Tokenize，遍搜文章Title及Comment是否有出現指定球星人名，加入對應球員專屬之字詞庫。

c. Word Token字詞過濾（Python）:

利用NLTK(Natural Language Toolkit)的詞性標記pos_tag功能將單一Token標注詞性，再將每一Comment內的單詞利用Word2vec計算字詞關聯性，將一段Comment內較不相關的名詞、連接詞做二次濾除。

d. 視覺化（Javascript & Python）:

使用上述處理完的所有資料，將球員專屬字詞庫生成文字雲的形式，利用網頁形式呈現，網頁同時也會去獲取相關球員的nba stats，目前我們選用球員的資料為歷年季後賽表現數據。

五、 分析結果

(Analysis results)

目前生成的文字雲僅針對巨星級球員，一是討論量夠多，生成的文字較為多樣，二是球員擁有的正負評價分佈較為平均。

以Stephen Curry之文字雲為例（16年例行賽MVP，三分球王）

當中分佈最多的為Good, Great, Bad, Wrong等較為常見之形容詞，也不乏粉絲給予 offensive(進攻性強), dribble(運球), underrated(被低估), professional(專業)，以及酸民給的ridiculous(荒謬), hilarious(滑稽)等等負面詞彙，其中也不少與比賽相關的評論，例如Curry某場進球率並不高，disappointed(令人失望)的討論度也會跟著提高，抑或是出現大三元時，triple double的出現頻率也會跟著提高。

估計只要我們將時間序列限縮並加入觀察，我們可以更有效地得到與比賽相關的球員評論。

Demo Site : https://nol.cs.nctu.edu.tw/~lincheyu/nba.html

六、 綜合討論 (Discussions)

這次Project比較偏向文字探勘與分析，但仍然遇上許多問題。

從資料收集層面來看，我們的資料較單純，比較沒有像其他組同學一樣需要normalization，但是仍然遇上Request數量限制以及回文格式常含有連結、meme等特殊字元的情況，在這步我們得先做第一次的過濾，不然會嚴重影響到我們後續去字與取字的方式。

從內容擷取來看，有時候標題並未含有任何球員資訊，可能只是隊伍、比賽，我們必須再深入留言挖掘並判斷關聯性字詞，才能得到想要的球員資訊、評價等資料，都是需要克服的痛點，最後還得從詞性再次過濾字庫，不同的詞性組合也會造成每個球員的評價不同。（e.g.若取名詞，常會出現隊友或是勁敵的字詞）

最後則是視覺化呈現，目前我們僅呈現固定球員（總冠軍賽明星球員）的資訊，未來希望能夠輸入球員名字即取得文字雲，並固定時間爬取reddit上之新資料，以達成實時分析之功能。

目前未實作Real-time輸入球員名字預測評價分析的原因是，在我們觀察完討論版生態後，發覺網民並不一定能夠根據球員實際表現給予球員評價，僅有巨星級球員能夠獲得較大量的討論，討論也包含了正反兩面評價以及酸言酸語（例如：某場打鐵，被酸到爆），這樣子的討論模式僅存在於明星身上。

再者，即使球員擁有相同身材，上場的位置也相同，仍會因為球隊戰術以及個人球風的不同造成極大的差異，故我們認為目前沒有一個比較好的方式去預測小眾球員的評價。

我們從這次Term Project的實作中學到了不少新知，以及了解資料科學對於產業的重要性，同時也認知到要當個資料科學家並沒有想像中容易。

參考文獻

1. Praw document: https://praw.readthedocs.io/en/latest/
2. Nltk document: http://www.nltk.org/
3. Spark-submit: https://spark.apache.org/docs/latest/submitting-applications.html
4. Apache Spark: https://spark.apache.org/