國 立 中 正 大 學

資 訊 工 程 研 究 所 碩 士 論 文

YouTube 影片分類統計瀏覽

Classified Statistical Browsing For YouTube Videos

研究生：賴麒竹

指導教授：吳昇　博士

中華民國　一零一　年　七　月

YouTube 影片分類統計瀏覽

研究生：賴麒竹　　指導教授：吳昇博士

國立中正大學資訊工程研究所

摘 要

隨著寬頻網路時代的來臨以及影片製作工具的普及，使用者不僅在網路上瀏覽影片，也經常在網路上分享影片。隨著線上影片數量越來越多，影片搜尋的困難度也越來越高。有時後使用者並沒有特定搜尋對象，此時如何提供一個良好的分類排行與導覽服務是一個重要課題。

在本篇論文中，系統利用分類詞彙資料庫的概念，設計一套影片自動分類機制，利用階層式分類概念，提供使用者可以方便的在感興趣的類別中瀏覽並觀看影片。並提供該類別下的詞彙統計資料，讓使用者可以快速地得知該類別下的熱門詞彙。

Classified Statistical Browsing For YouTube Videos

Student: Chi-Chu Lai Advisor: Dr. Sun Wu

Department of Computer Science and Information Engineering National Chung Cheng University

Abstraction

As the available bandwidth for each user increases and the popularization of video-making tools, users not only browse videos online but also share videos online. With the amount of online videos increasing, searching for wanted videos become more difficult. Sometimes, the user may not have specific keywords to search for. In such circumstances, providing a convenient classified browsing with statistical information becomes an important issue.

In this thesis, we use the concept of classified term database to design a mechanism for automatic classification of videos. The purpose of the hierarchical category system is to make the user more convenient in browsing and watching videos of user's interest. In addition, we provide statistical data for each category, so the users can get the information of popular terms in the category.

誌 謝

一眨眼，六年的中正生活也隨著這篇論文畫下一個句點。首先，感謝我的指導教授 — 吳昇老師，不僅是在學術研究領域上的教導，更多的是提供不同的角度看待生活中許多事情，讓我對於很多事情了新的看法。也感謝實驗室內的每位學長姊、同學、學弟們，在兩年的研究所生涯裡，在我遇到困難時，不吝惜的伸出援手，幫我解惑，另外，也帶來許多歡笑，使研究所的生活不那麼沉悶。

感謝一直陪伴我的所有朋友們，雖然都已各奔東西分散在各地，無法常常相聚，但大家仍保持聯繫，分享彼此的生活點滴，且每當我心情不好時，你們仍願意當我的垃圾桶，聽我吐苦水，陪我一起抱怨；在高興興奮時，聽著我的胡言亂語，陪我一起分享，謝謝你們。

最後，也是最重要的，感謝我的家人們，謝謝父母給予我經濟上的支援與生活上的照料，讓我可以全心全力專注於學業上，也謝謝你們，在我情緒低落時，無條件包容我的負面情緒。謝謝姐姐們包容我的任性，和妳們鬥嘴是我生活中很重要的樂趣。也謝謝我的外甥小朋友們，每次看到你們天真無邪的笑容、逗趣的表情動作與對話，不僅讓我拋開煩惱，也讓我重新找回堅持下去的動力。謝謝我所有的家人們，謝謝你們。

目 錄

圖表目錄

表格目錄

Chapter 1. Introduction

隨著網路頻寬不斷的提升與影音串流技術的進步，資訊的分享不再侷限於靜態的文字與圖片，動態的影音內容呈現已逐漸成為趨勢，在此趨勢下，網路影音觀賞也已成為網路使用者的休閒活動之一，各式各樣的影音分享網站應運而生。

根據網路影音服務 Wistia[1] 的研究調查，2006~2011 年間，網路影音使用人口從 33%、48%、52%、62%、66% 逐步成長至 71%；每天都會造訪線上影音網站的比例也由 8%、15%、16%、19%、23% 一路成長到 28%。

隨著影音分享風潮的盛行，許多影音網站也隨之出現。目前排行第一的影音分享網站 — YouTube[2] (Table 1)，於 2011 年的統計數字中，全部使用者每分鐘上傳到 YouTube 的影片長度加起來約 60 小時，相當於每秒鐘就有長達 1 小時 (60 分鐘) 的影片上傳到 YouTube；全世界使用者每天觀看的影片超過 40 億次。如果影片平均長度以 6 分鐘計，一天上傳量達 88 萬部，一年就達三億部以上，也就是說 YouTube 上的影片總數應該有超過十億部。而如此龐大的影片資料中，也衍生出一項問題：使用者如何篩選出符合自己需求的影片？

目前大多數的影音網站皆提供依照各種統計數據 (例如: 瀏覽次數、推薦次數、上傳時間)，推薦使用者熱門影片觀看。部分影音網站另提供影片分類的功能，透過影片上傳者對影片資料欄位的設定，對影片進行分類，讓使用者可以瀏覽該類別下的所屬影片。

依照使用者對影片種類的設定所提供的分類瀏覽功能，潛藏著一個最大的問題是：若影片上傳者未設定類別，或是設定錯誤呢？而且現今多數的影音網站皆

Rank	Site	Unique Visitors (users)	Reach	Page Views	Has Advertising
1	YouTube.com	800,000,000	46.8%	100,000,000,000	Yes
2	youku.com	140,000,000	8.2%	4,000,000,000	Yes
3	tudou.com	110,000,000	6.3%	2,700,000,000	Yes
4	pps.tv	60,000,000	3.5%	500,000,000	Yes
5	soku.com	45,000,000	2.6%	790,000,000	Yes

Table 1. 2011 熱門影音分享網站

(Data From: Google Doubleclick3)

只提供影片一種種類的設定，因此當影片可分類至兩者以上的種類時，其他使用者只可在影片上傳者設定的類別中查看該影片，而無法在另一類別中看到。而除了大方向的影片分類外，子分類亦是另一潛藏的問題，倘若使用者只想得知該分類類別下的某個子分類種類，例如：音樂種類下的古典音樂，那此時使用者又該如何取得自己想要的影片呢？尤其在該類別下隱含某種主流子類別時(例如：音樂類別中，以流行音樂為主觀看項目)，其他較小範圍的子類別影片便不易找出，甚至當兩種子分類項目有某種程度的衝突感時，例如：流行音樂與古典音樂、恐怖電影與兒童電影，造成使用者在某大類別搜尋所需影片的同時，必須不時的瀏覽到自己排斥的影片，如此一來，分類的益處不僅無法達到，反而造成使用者的困擾。

因此本論文希望提供一個影片瀏覽系統，可自行分類影片並提供更為詳細統計資料，讓使用者一方面可得知依照過去某段期間 (日/週/月) 的瀏覽次數、喜愛次數的統計值，並將該段期間的統計值與該統計期間之前的平均統計值作比較，取得其上升或下降的情況；除了全部影片的統計資料外，系統也會提供特定分類項目的分類影片瀏覽，並同時提供該分類項目的統計資訊；若是系統預設的分類

方式並不是使用者預期的分類方式，使用者亦可藉由自行提供的分類詞彙，取得想要的分類影片。

本論文編排架構如下。在第二章中，將簡單介紹目前相關的影音分享網站的分類技術以及本系統實作時的相關背景資料；在第三章中，會闡述本系統的架構與實作方式；在第四章中，介紹本系統的實作介面與功能說明；第五章為結論與未來展望，最後第六章則為系統開發過程中參考之資料來源與套用的套件。

Chapter 2. Related Work

2.1 Related Video Websites

本小節將會簡單介紹兩個現有影音網站現況與其熱門推薦和分類方式。

2.1.1 YouTube

YouTube 自 2005 年成立，2006 年被 Google 收購，至今日已為排行第一的影音分享網站(Table 1)。每天被觀看影片自 800 萬影片(2005/12)，成長至 40 億以上(2012/01)。

YouTube 影片共分 15 種類別，類別設定須於使用者上傳影片時，自行設定一種分類。15 種類別分別為：汽車與交通工具、搞笑、教育、娛樂、電影與動畫、遊戲、DIY 教學、音樂、新聞與政治、非營利組織與行動主義、人物與網誌、寵物與動物、科學與科技、運動、旅行與活動。

YouTube 的影片推薦功能可分兩種，除了在各分類類別下，依照各種影片數據對特定時間(Table 2)的統計資料進行熱門排行；另外也提供依照使用者的歷史瀏覽紀錄，推薦使用者可能有興趣的影片 。

統 計 對 象	對 應 時 間 週 期
評論數目：	指定過去一年的任意一週
使用者喜歡數：	每日／每週／每月／不限時間
HD 高畫質影片的觀看次數：	每日／每週／每月／不限時間
觀看次數：	每日／每週／每月／不限時間
受喜愛數：	每日／每週／每月／不限時間

Table 2. YouTube 排行榜設定條件

2.1.2 Youku

Youku[4] 於 2006 年成立，2010 年於美國紐約證券交易所掛牌上市，2011 年根據 comScore[5] 調查，10 月瀏覽量為 46 億人次，為 2.3%全球市佔率，2012 年與中國大陸另一家知名影片分享網站 — Tudou[6] (土豆) 進行合併，使其不僅是全球網站流量排名第二的影片分享網站，更在中國大陸達到超過八成的網路視訊市占比例。

Youku 影片分類可分為一級分類與二級分類，一級分類目前共含 20 種 (資訊、電視劇、電影、綜藝、音樂、動漫、紀錄片、體育、旅遊、汽車、科技、原創、遊戲、搞笑、生活、時尚、娛樂、教育、母嬰、廣告)，每種一級分類下 (廣告類別除外) 可再依影片型態細分出二級分類 (例如：音樂類別可再細分為流行、搖滾、舞曲、電子、R&B、HIP-HOP、鄉村、民族、民謠、拉丁、爵士、雷鬼、新世紀、古典、音樂劇、輕音樂，共 16 種子類別)。一級分類須於使用者上傳影片時，自行提供設定，上傳後於系統審核影片階段，再由系統判定，設定二級分類，且審核階段時，會再次判定使用者設定的一級分類有無錯誤，若系統判定分類有誤，則會通知使用者，拒絕該影片正常發佈。每部影片只可指定一種一級分類，但二級分類最多可設定三種。

Youku 於每種一級分類下(財經種類不提供)，可搭配二級分類，於特定時間範圍內(日/週/月/全部)，依照發佈日期/播放次數/評論筆數/被引用次數/被收藏次數/爭議數等排序條件，提供影片排行。

2.2 Background Tool

2.2.1 MMonkey

MMonkey Tool 由 C 語言實作，為 GaisLab 指導老師 吳昇博士開發。MMonkey Tool 提供多重詞彙比對搜尋，可找出字串組（含一或多個字串）中任一字串在另一指定字串中出現的位置。另可設定欲找尋的字串組的比對條件：case insensitive / case sensitive,，word / substring match。本系統底層使用的多重詞彙比對搜尋功能，即是呼叫 MMonkey tool 進行實作。

2.2.2 Gais Record Format

Gais Record 為 Gais Lab 常用的資料儲存格式，為純文字內容。依照使用者自行定義的 Record Delimiter（例如：@\n@id:）分隔每筆 Record，並自行定義所需的欄位，由 Field Tag 區分各欄位儲存內容，Field Tag 格式多為「\n@」開頭、「:」結尾，如「\n@title:」。Gais Record 範例可見 Table 3 與 Table 4。

```
@
@id:-12a410cqdI
@title:No Problems
@keyword:Problem
@
@id:ab01t6haEva
@title:Video – Cute Dog
@keyword:Dog
```

Table 3. Gais Record 範例

	Record 1	**Record 2**
id	-12a410cqdI	ab01t6haEva
title	No Problems	Video – Cute Dog
keyword	Problem	Dog

Table 4. Get Record 範例欄位 From Table3

(Record Delimiter：「@\n@id:」)

Chapter 3. System Design and Architecture

3.1 System Architecture

本論文主要目標為提供一個能讓使用者分類瀏覽,並針對指定對象（例如：歌手、歌曲、電視節目）提供數據變化統計的影片分享平台。

如 Figure 1 所示，為達到分類瀏覽的功能，本系統實作一個 Term DB Based 的分類機制，透過對每個分類進行 Term DB 的設定，以 Term DB 作為影片分類的主要篩選條件，利用 Multi-Query Machine，當 Term DB 內的 Term 出現在影片文字內時，即可判定該影片屬於該分類。

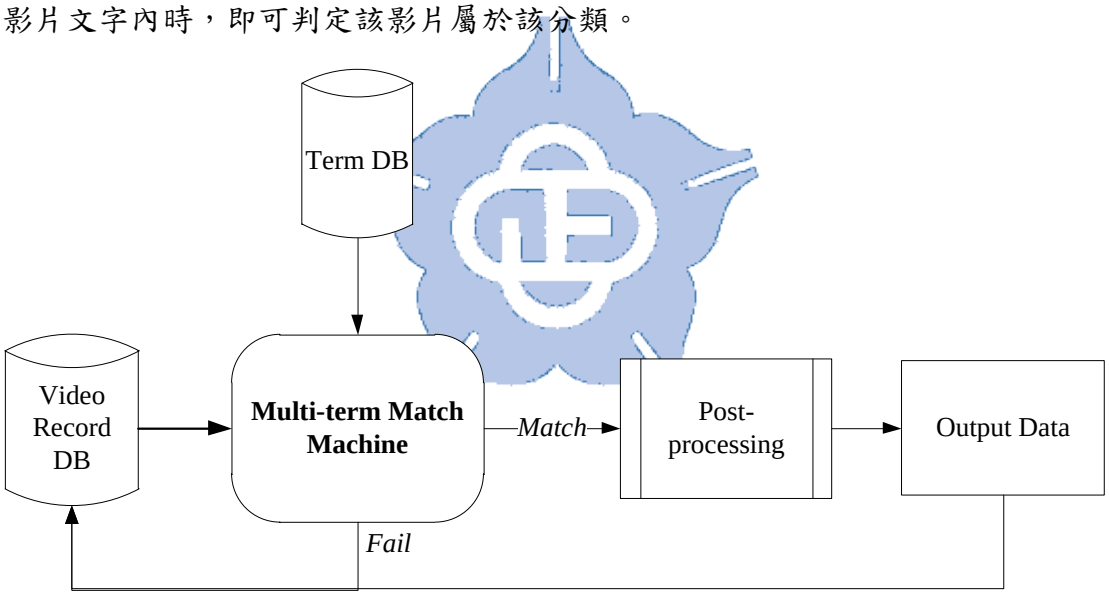


Figure 1. 系統內部架構

經過系統底層 Term DB Based 的分類機制後，對於符合該類別的影片，系統將視各階段所需條件對影片做後續處理，舉例而言，進行詞彙數據統計時，對於符合該分類的影片，系統將會保留該影片的觀看次數、喜愛次數等欄位資料進行統計；若是要對影片加上分類標籤，則是完整保留原始影片資料，並在最後加上標籤欄位，以及此影片所屬的分類標籤。處理後的影片資料，則再依系統呈現

格式輸出，提供使用者於介面瀏覽。關於影片後續處理的詳細內容將於 3.4 節詳細介紹。本系統整體其架構如 Figure 2 所示。

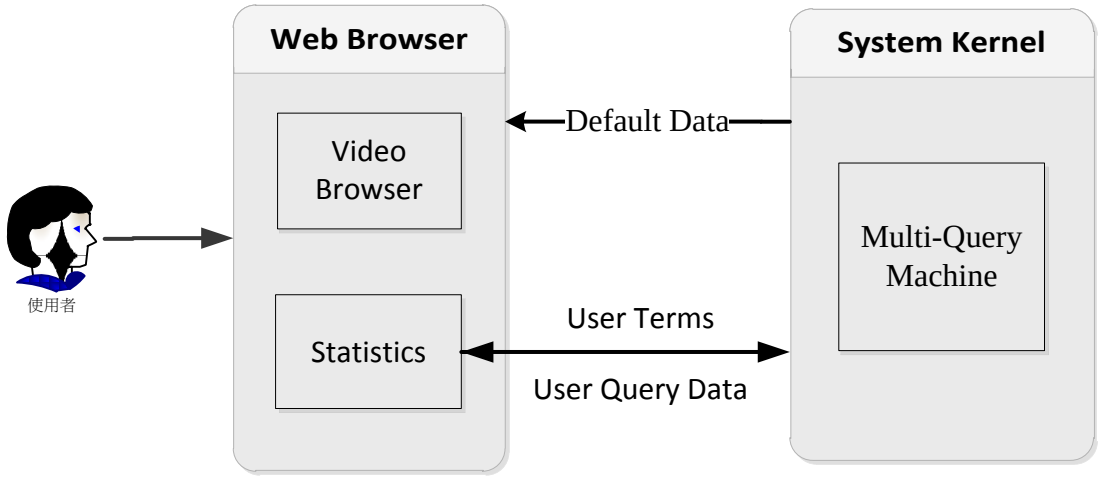


Figure 2. 系統整體架構

3.2 Preprocess — Data Set

3.2.1 Term DB

Term DB 為本系統對影片分類的主要篩選條件，因此每個分類的 Term DB 必須是可以完整代表該分類內容的詞彙集合。例如：汽車分類的 Term DB 主要為各個汽車廠牌名稱 (如：福特)，與其名下的汽車款型 (如：Audi A6)；3C 分類則為各個 3C 產品或廠商名稱 (如：iphone、Asus)；流行音樂類別則為各個歌手名稱所構成。

為方便最後的分類正確確認，本系統現階段將分類對象著眼於較為熟悉的分類 — 流行音樂。流行音樂類別中主要的構成詞彙 — 歌手名稱，擷取自魔鏡歌詞網[7]，並依照該網站所設的流行音樂類別：華人男歌手、華人女歌手、華人團體歌手、日韓歌手、西洋歌手，對流行音樂類別提供子分類影片瀏覽，各子類別所含的詞彙數量如 Table 5 所示。

	華人男歌手	華人女歌手	華人團體歌手	日韓歌手	西洋歌手	總和
原始詞彙數(未刪除同人不同名的歌手名)	2038	1982	1036	2860	5182	13,098

Table 5. 各分類原始詞彙數目

3.2.2 Video DB

本系統自 2012-02-28 開始，固定於每日從 YouTube 抓取影片資料。在 Term DB Based 的 Multi-Query Match 機制下，系統對於影片分類的判斷主要著眼於影片資料的文字描述內容，因此在相信 YouTube 的搜尋機制可以提供使用者最相關且最熱門的 Video 資料前提下，可以假設透過 YouTube Search，系統將取得所需之影片資料。意即將 Term DB 內每筆詞彙設為 Search Keyword，對 YouTube 進行 Query Search，YouTube Search Results 即為系統所需之影片。

大多數情況下，使用者搜尋資料時往往只會查看最前面的搜尋結果，一旦找尋到所需之資料後，排在後方的搜尋結果便不再瀏覽，因此網頁流量越少，網頁內容對使用者需求符合機率便會相對較低，因此在相同的相關度條件下，搜尋引擎大多會將熱門網頁排在搜尋結果前面。相同的，影片搜尋往往會將與關鍵字最相關且最熱門的影片排在搜尋結果前面。

一方面，本系統欲提供的分類頁面瀏覽，正常情況下，使用者大多期盼看到該類別下最熱門的影片；另一方面，數據越大的影片，在統計數據變化時，往往會有越大的影響。因此，可以預期以 Term DB 為 Search Keyword 所取得的搜尋結果中，利用排在最前面的 N 筆影片進行處理，即可大致達到本系統欲達到

的系統目標。系統目前將 N 設為 100，即每筆 Term Query 保留最相關的 100 筆影片資料。

利用 YouTube API[8]，可輕易執行 YouTube Search，如 Table 6 所示，透過發送 Get Request 至 YouTube API URL，設定所需之參數值，即可取得搜尋結果。由於每筆 Term 需紀錄 100 筆相關的影片資料，受於 API 參數「max-results」最大值為 50 的限制，因此每筆 Term 須執行兩次 URL Request，每個 Query Term 執行的 URL 格式將如 Table 7 所示。

參 數	定 義	有效值	預設值
alt	搜尋結果的回傳格式	atom 、 rss 、 json 、 json-in-script	atom
max-results	回傳結果的最大筆數	0 ~ 50	25
start-index	指定搜尋結果的偏移筆數，即自原搜尋結果第 N 筆開始，回傳其下面 max-results 筆影片	1 ~ ∞	1
v	YouTube 處理 API 請求時所用 API 的版本	1、2	1
q	指定搜索查詢字詞	不限	空字串

Table 6. YouTube API v2.0 – API 相關參數

Index	URL
1 ~ 50	https://gdata.YouTube.com/feeds/api/videos?q="*term*"&max-results=50&start-index=1&v=2&alt=json
51 ~ 100	https://gdata.YouTube.com/feeds/api/videos?q="*term*"&max-results=50&start-index=51&v=2&alt=json

Table 7. YouTube API 套用範例

由於系統欲提供流量變化的資訊，必須取得影片每天的數據值。Term DB 總共含 13,098 筆 Key term，每筆須執行 2 次 URL Request，每天所需執行的 URL Request 約 26196 次 (Term DB 持續更新中，因此每天執行的 URL Request 量也逐漸成長中)，為了縮短每天抓取的時間，系統將所有 URL Request 分散於三台 Server，每台 Server 每天約執行 8800 次 URL Request，平均每台每次抓取約執行 30 分鐘。

由於自 YouTube 取得的影片格式為 json 型態，因此系統必須將資料轉為系統使用的 Gais Record Format (見 2.1.2 小節)。Table 8 將會介紹底層使用的完整資料欄位與範例內容。

此外，系統為了提供影片每 日/周/月 的流量變化，在 parse json record 時，會同時建立一份以 Video ID 為 key，viewcount 和 favoritecount 為 value 的 key-value hash table，使系統底層在計算流量變化時，可快速地利用 hash function 找出該影片於一日/周/月 前的統計值，不須再於資料量大的完整影片資料內，重新呼叫 Multi-Query Match Machine 找出詞彙存在的影片資料。

因此，影片資料最終會以三種形式儲存於系統內，第一種形式為自 YouTube 取回最原始的資料 (json 格式)；第二種為轉換為 Gais Record Format 的資料格式；第三種則為簡易以 Video ID 為主的 key-value hash table。

欄位名稱	欄位內容	範例內容
@id:	影片 ID (由 YouTube 設定)	vdiGFBEY9T0
@published:	影片上傳時間	2012-03-23T15:09:07.000Z
@updated:	影片更新時間	2012-06-08T03:03:24.000Z
@title:	影片標題	Westlife-The Rose cover by Jocelyn
@content:	影片描述內容	Jocelyn age 12 singing - The Rose by Westlife
@author:	影片上傳者	jjparents
@keyword:	由 YouTube 抓取時，所設定的 Term Query (由本系統自行設定)	Jocelyn, westlife, 西城男孩
@favoriteCount:	喜歡次數	4
@viewCount:	觀看次數	3494
@duration:	影片長度(秒)	210
@category:	YouTube 預設的影片類別	Music
@class:	系統經過 Multi-term Match Machine 後，所判斷的該影片分類	music, pop, WomanSinger,

Table 8. Gais Record Format

3.3 Multi-Query Machine

系統核心會利用屬於該分類類別下的 Term DB，找出含有 Term DB 內部詞彙的影片，對於符合該分類的影片，再進行後續處理，處理流程如 Figure 3 所示。

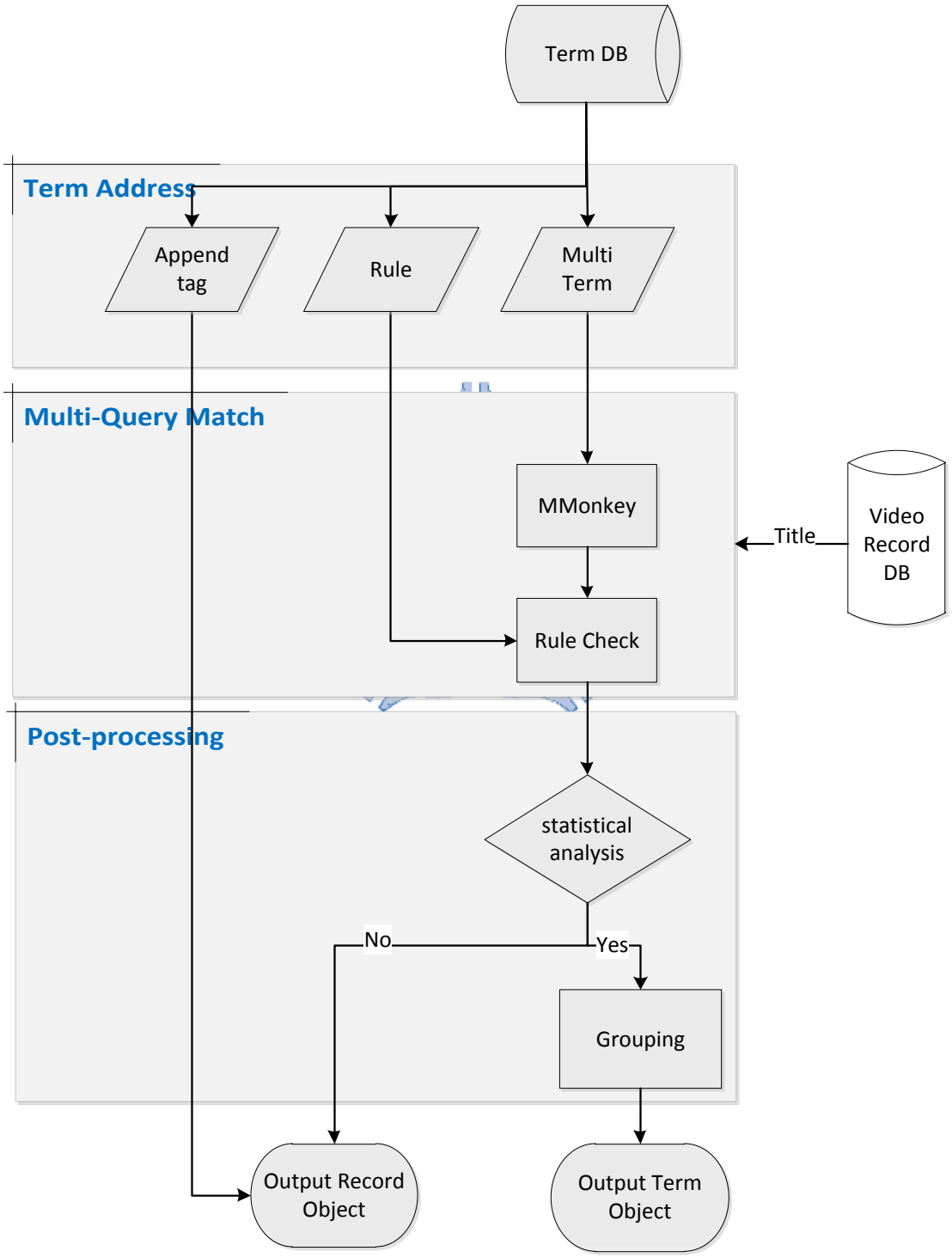


Figure 3. Multi-Query Machine 架構

在 Term DB Based 的分類概念下，系統希望可以透過詞彙在影片文字內的出現與否作為分類的依據，在此機制下，影片文字與詞彙即為整個機制下最主要的兩大元素，因此為了可以提供好的呈現結果，系統會在此兩部份分別設定條件篩選。

3.3.1 Video Match Text

一筆 YouTube 影片資料中含有許多不同欄位，包括影片標題、影片內容描述、標籤、種類、影片上傳者等相關資訊，其中影片上傳者是跟影片內容無相關的欄位；種類則是 YouTube 設定的 15 項分類名稱，其分類名稱可代表的意涵過度廣泛，因此無法用來作為找尋詞彙出現的欄位位置。

標籤則是由使用者自行設定，通常為可代表該影片的相關詞彙，但此部分的設定，一方面，受限於需交由使用者自行設定，因此設定的多寡會影響詞彙比對的結果，另一方面，經過觀察，標籤的內容多為影片標題的部分內容字串，即標籤多是使用者認為在影片標題中可簡單描述影片內容的詞彙，故當標題已無法正確描述影片內容時，其標籤通常更無法達到正確形容影片內容的功效，因此使用影片標籤作為判斷的對象，不只受限於標題，亦會受限於使用者設定，故其效果不如直接使用影片標題作為判斷內容的情況佳。

影片內容含有詳細的影片內容描述，但由於太過詳細，往往會造成雜訊而干擾影片的判斷，舉例而言，一首歌曲的描述內容，除了影片的介紹外，可能還包含歌詞，而歌詞則會含有太多詞彙，且多不直接關聯於影片內容，因此，既無法有效達到增加分類的正確性率，又可能會造成其他 Query 的錯誤判斷。故我們最後在判斷詞彙出現在影片中的位置時，只會選擇詞彙有出現在影片標題的影片。

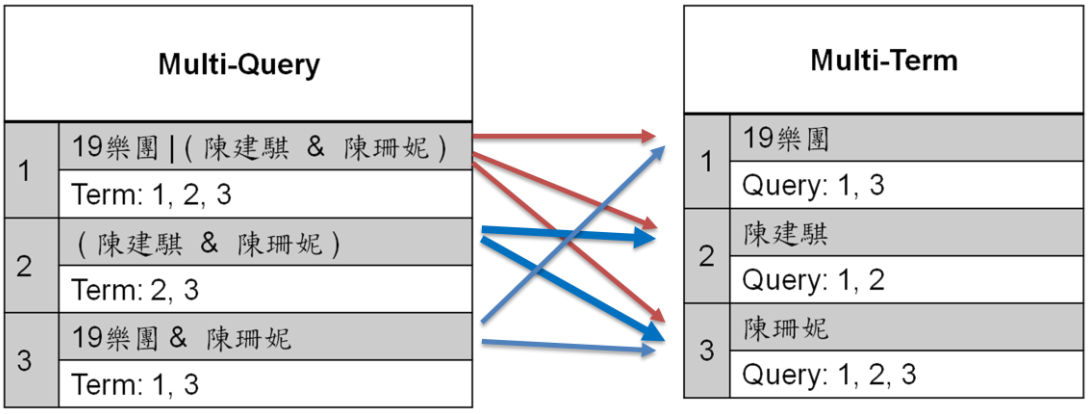
3.3.2 Multi-Query

在分類下定義屬於該分類的 Term DB，透過影片標題含有該詞彙與否而給予分類判斷，此方式是系統最主要的核心概念，但此方式卻隱含一些問題，包括歧異詞彙、同義詞彙等問題。舉例而言，『Sweety』可同時代表台灣團體歌手，也可代表是日本歌手，因此，在找尋日本歌手『Sweety』時，我們必須多加確認含有『Sweety』字串的影片標題，其整體內容是否是代表為日本的歌手。而有時候同一個代表對象卻會有不同的詞彙可以表示，舉例而言，『westlife』與『西城男孩』同樣代表同一個愛爾蘭男子歌唱團體，可是我們若只下一個詞彙時，可能就會造成以另一個詞彙表示的影片無法正確被找尋到。

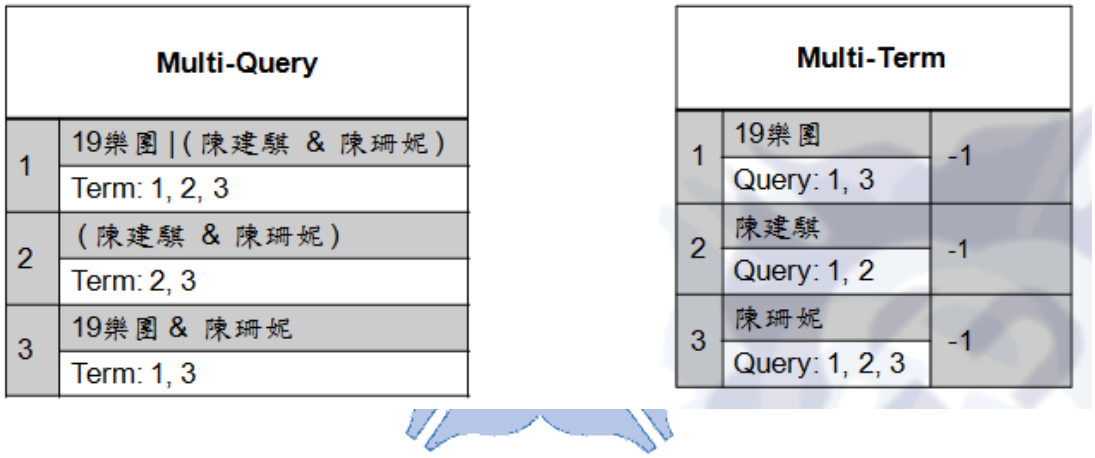
因此，基於以上的潛在問題，系統將 Multi-Term Match Machine 進一步修改為 Multi-Query Match Machine。欲找尋某物件存在的影片資料時，不再受限只能用單一詞彙涵蓋全部意涵，而可以使用多個詞彙搭配 AND / OR 條件，達到物件對象正確完整的描述。舉例而言，可以使用 Query：『劉承燦 | 刘承灿 | 柳承灿 | 유승찬』代表同一個韓國歌手的不同翻譯名字。

在 Multi-Query 與 Multi-Term 的轉換過程中（Table 9），系統會對各自建立一份 Mapping Table，並各自設定索引資訊。每個 Query 會記錄其所含的 Term 的相對索引，而每個 Term 也會記錄其原始存在的 Query 索引，並在 Multi-Term Table 內對每個 Term 紀錄最後一次出現的影片索引位置，因此當 Term 有符合情況時，利用 Term Table 內記錄的原始存在之 Query 索引，找出需判斷的 Query 筆數，Query 再進一步利用 Query Table 內記錄的所含 Term 索引資訊，搭配該 Term 紀錄的最後一次出現的影片索引位置，逐一判斷是否符合 Query 條件。

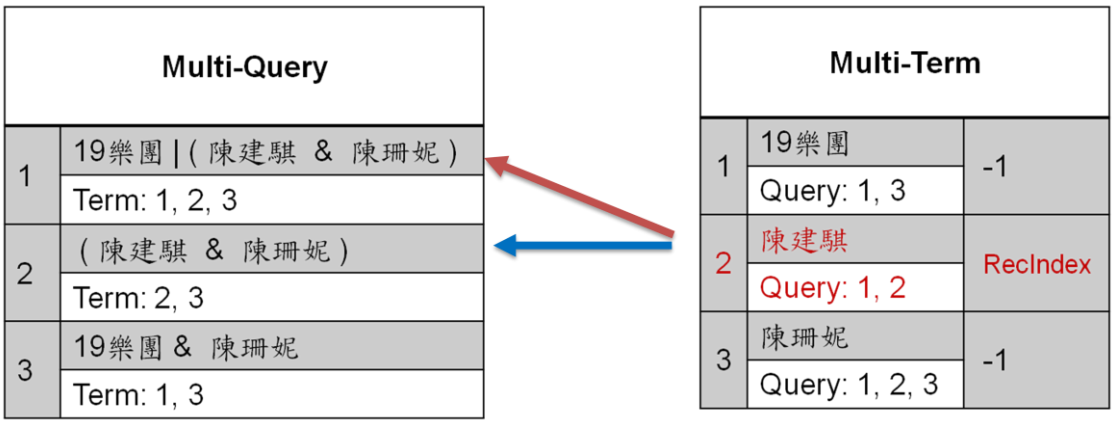
Step 1. Multi-Query →Multi Term



Step 2. 初始化 Multi-Term Table 的影片比對索引紀錄



Step 3. 將比對到的 Term 設定索引紀錄，並推回 Query Index



Step 4.存在該 Term 的 Query 逐一確認其 AND / OR 條件是否符合

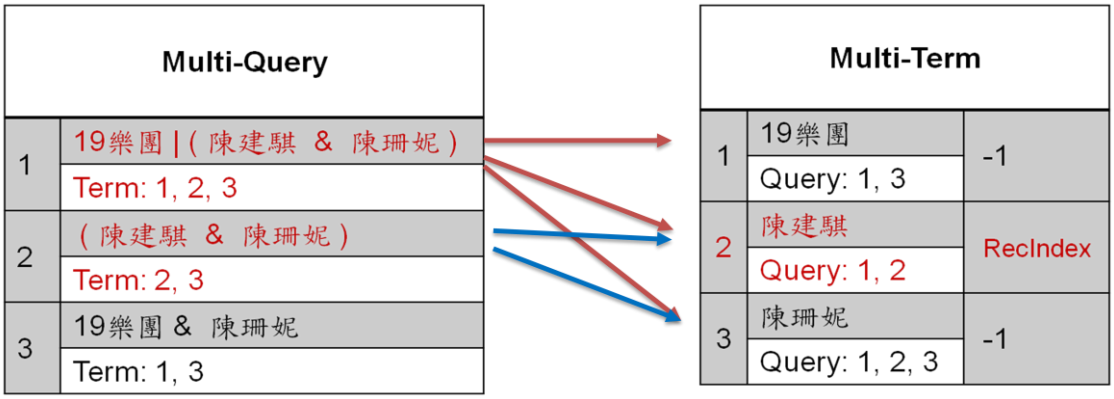


Table 9. Multi-Query 與 Multi-Term Mapping 流程

3.3.3 Multi-Term Match

本系統核心概念即是透過 Query 內 Term 的搜尋比對達到自動分類的效果，因此系統內部會呼叫 Multi-Term Match Tool 進行字串比對搜尋，此部分工具系統是使用『MMonkey Tool』，此工具的詳細介紹請見 2.2.1 節。

3.3.4 Rule Filter

在經過 Multi-Term Match 以及 AND / OR 的多重條件下，系統依然存在潛在問題，最主要因素在於，儘管使用 Multi-Term Match 搭配 AND / OR 可以達到全面性的描述 Query Object，但除非該 Query Object 含有相當代表性的關鍵詞彙，否則，此 Query Object 必須搭配相當多的詞彙才可完整的描述該 Query Object，尤其當此 Query Object 名稱是屬於相當普遍存在的詞彙時，更容易造成篩選上的錯誤情況。因此，在此情況下，系統需要另一層篩選機制，提供更全面性的篩選，降低錯誤情況的發生。

舉例而言，『Live』為一組西洋樂團，當需找出此樂團的影片時，單純使用『Live』作為 Query Term 是不足的，因為『Live』同時代表現場演出的意思，因此許多屬於現場表演錄影，或是現場轉播的影片，其標題也同時會出現此詞彙。因此，若須找出此樂團的影片時，必須搭配其他篩選機制，若使用 Multi-Term Match 與 AND / OR 的方式，其中一種作法為，除了該樂團團名外，再加上該樂團所有歌名，方可正確找出該樂團的影片，但使用此法會造成此 Query Object 相對龐大，且還需額外擁有樂團的歌名資訊，因此，在此情況下，Multi-Term Match 搭配 AND / OR 的方式無法發揮其效用。

為了彌補此部分的不足，我們透過觀察標題格式，得到一種全面性的篩選規則。在大多數情況下，當影片純粹是一首歌曲的影片(如：MV)時，歌手與歌名往往會使用特殊符號隔開，表示彼此的相對應關係，例如：「Justin Bieber - Boyfriend」，即代表歌手為「Justin Bieber」，歌曲為「Boyfriend」。因此當無法在純粹使用 Query 時，篩選出正確結果時，我們可利用此種較為嚴格的篩選條件，找出屬於該歌手的歌曲。

此外，部分標題上還會顯示該歌曲的相關資訊，例如：演唱地點、演唱時間、實況轉播等相關資料，同樣的，為了區隔彼此間的對應關係，各資訊仍會以特殊符號隔開，因此，在利用標題格式區分影片型態時，可利用特殊符號區隔開的字串數，降低影片符合條件標準，並搭配匹配詞彙出現的位置多一步確認。目前系統提供的相關規則篩選可見 Table 10。

規則	範例	代表意涵
Delimiter	‘/’,’,’	以斜線(’/’)和逗點(’,’) 兩種字元切割標題文字。欲找尋的詞彙須包夾在此兩種字元間。
Delimiter String Count	‘maxwords=4’, ‘minwords=2’	以特殊字元切割出的字串數最多只能含 4 個，最少則須 2 個。
Term Match Position	‘/^/’,’/$/’,’/^$/’	欲找尋的詞彙在符合特殊字元包夾的條件下，仍需符合在標題在起頭處(‘/^/’)，結尾處(‘/$/’)，或起頭與結尾任一位置皆可(‘/^$/’)
Simple Language Check	‘English’,’-English’	簡易判斷標題內容是否要全英文。

Table 10. 詞彙相關篩選規則

3.4 Post-processing

在經過 Multi-Query Machine 後，必須對符合該類別的影片資料進行後續處理，提供系統呈現時所需之內容。

3.4.1 Append Field Tag

對每筆原始影片加入「@class:」欄位，並執行 Multi-term Match Machine，將屬於該分類的影片在「@class:」欄位裡面加入該分類的標籤，每部影片的分類標籤數目不限制，目前系統內設定的標籤內容共含 7 種，於 Table 11 呈現。

分類類別	標籤
音樂	Music
流行音樂	Pop
華人男歌手	ManSinger
華人女歌手	WomanSinger
華人團體歌手	GroupSinger
日韓歌手	JKSinger
西洋歌手	WestSinger

Table 11. 系統內之分類標籤

3.4.2 Extraction Simple Video Data

為了提供使用者可以更快速的頁面分類瀏覽，對於屬於該分類下的影片將會如 Table 12 所示，只記錄該影片重要的數值變化量與該影片在完整影片資料檔案的所在位置 (便於快速取得該影片完整資料)，將所有該筆影片的數值以空格區分放置於檔案內同一行。一方面，可簡化前端頁面的處理過程，使頁面端在依照特定欄位值排序影片時，不需先將所有的影片進行欄位切割，找出欄位位置與欄位內容，只需將每行 (即每筆影片) 依照空格快速切割後，便可得其欄位數值，依其數值進行排序，欲輸出影片於網頁時，則只需依照簡易記錄中的該影片所在位置，至完整影片資料檔案中的該位置便可取得該影片內容；另一方面，網頁端不需讀取完整影片資料內容，只需讀取檔案小的影片簡化數據檔，節省記憶體的使用。

範例 ： **(One Line)**	**s4Ajtk743fY 195938 1433 17858 120689 1395 8 106 924 34 4082 483280726**
各數值定義：	PID 與 其相關 11 項數值 1. Video id：s4Ajtk743fY 2. 影片總觀看次數：195938 3. 影片昨日觀看次數：1433 4. 影片上週觀看次數：17858 5. 影片上月觀看次數：120689 6. 影片總喜歡次數：1395 7. 影片昨日喜歡次數：8 8. 影片上週喜歡次數：106 9. 影片上月喜歡次數：924 10. 影片已發佈天數：34 11. 影片長度：4082 12. 影片於完整資料檔內的位置：483280726

Table 12. 簡化影片數據格式

3.4.3 Compute Term Ranking

為了計算每個分類類別下的 Term DB 每 日/周/月 的數據變化，系統將會對每項分類執行 Multi-term Match Machine，計算該分類所屬之 Term DB 內每筆 term 在固定期間內 (日/周/月) 出現的影片數目、總觀看次數、總喜愛次數，並保留該 term 出現的影片中，觀看次數增加最多的五筆，記錄此五筆影片的 ID、標題、增加的觀看次數與喜愛次數數量。

除了計算在最近一 日/周/月 內的變化，系統也會統計自 2012-02-28 (本系統 Video DB 內最初收集 Video Data 的日期) 至該統計週期前一天的此過去期間內，該 term 出現的影片數目、總觀看次數、總喜愛次數，可用於觀察此 term 的數據變化情況，得知該 term 在近期內數據是否為正常消長，或是有大量的變化，可間接取得近期內爆紅的詞彙。

經過統計後的資料會以 json 格式輸出，每個 term object 會包含 9 個 key-value 集合，詳細內容於 Table 13 列表呈現，輸出範例可見 Table 14。

key	Value define
term	查詢的詞彙 (term)
ratio	Term 在近期時間週期的觀看次數與過去平均觀看次數變化
Ovideo	Term 在過去期間出現的影片數量
OsumV	Term 在過去期間的平均觀看次數
OsumF	Term 在過去期間的平均喜歡次數
video	Term 在近期時間出現的影片數量
sumV	Term 在近期時間週期的觀看次數
sumF	Term 在近期時間週期的喜歡次數
Top	Term 在近期時間週期觀看次數增加最多的五筆影片，每筆影片各含 ID、Title、viewcount、favoritecount

Table 13. Term 統計結果之輸出欄位

```
{
   "term":"Westlife",          "ratio":"1.08875",
   "Ovideo":"91",    "OsumV":"747201",      "OsumF":"1253",
   "video":"91",      "sumV":"813515",       "sumF":"1614",
   "Top":{
      "Top1":["ulOb9gIGGd0", "Westlife - My Love ", "97154", "96"],
      "Top2":["Rkkw8RhH9ck", "Westlife - You Raise Me Up (With Lyrics) ",
      "79848", "184"],
      "Top3":["WHyzxVlOI98", "Westlife - My Love (With Lyrics) ", "64462",
      "91"],
      "Top4":["7NrQei36fJk", "Westlife - If I Let You Go ", "39931", "48"],
      "Top5":["QWedHjSsjZA", "Westlife - Uptown Girl with lyrics ", "31597",
      "70"]
   }
},
```


Table 14. Term 統計結果之輸出範例

Chapter 4. System Interface

4.1 Classified browsing

網頁主要以分類瀏覽的方式觀看各類別的影片內容，介面如 Figure 4 所示，網頁各區於 Table 15 介紹。

Figure 4. 分類瀏覽介面

區域代碼	區域定義	呈現內容
1	標籤	目前此瀏覽頁面的分類標籤定義
2	類別	以階層式的概念顯示各類別間的相互關係
3	排序條件	設定下方影片呈現的排序依據。 分別為：總觀看人次、昨日觀看人次、上週觀看人次、上月觀看人次、影片發佈日期、總喜愛人次、昨日喜愛人次、上週喜愛人次、上月喜愛人次、影片長度由短至長排序、影片長度由短長至短排序，共１１種排序依據。
4	統計資料	顯示該類別下的 Term DB 的統計資料
5	影片瀏覽	瀏覽該類別的影片資料
6	使用者自訂	由使用者自行定義所需統計的詞彙資料。

Table 15. 介面各區介紹

4.2 Statistical

此部分將呈現依照分類類別下的 Term DB 進行分群統計的資料。資料內容包含 昨日/上周/上月 三種統計期間的資料。

資料呈現的方式可選擇圖表 [9] (Figure 5) 或列表 [10] (Figure 6) 呈現，圖表呈現部分主要呈現熱門的詞彙觀看次數相對情形，因此最多只會呈現前 100 筆觀看次數最多的詞彙。詳細資料則可藉由列表方式呈現，列表除了顯示該歌手於該期間的總觀看次數與總喜愛次數，另包含於過去期間該歌手的平均觀看人數、統計期間與過去期間的觀看人數變化、以及該歌手觀看人數最多的影片標題，若欲察看該歌手觀看次數最多的前五大影片，可點選該歌手名字，即會在列表右方呈現 (Figure 7)。

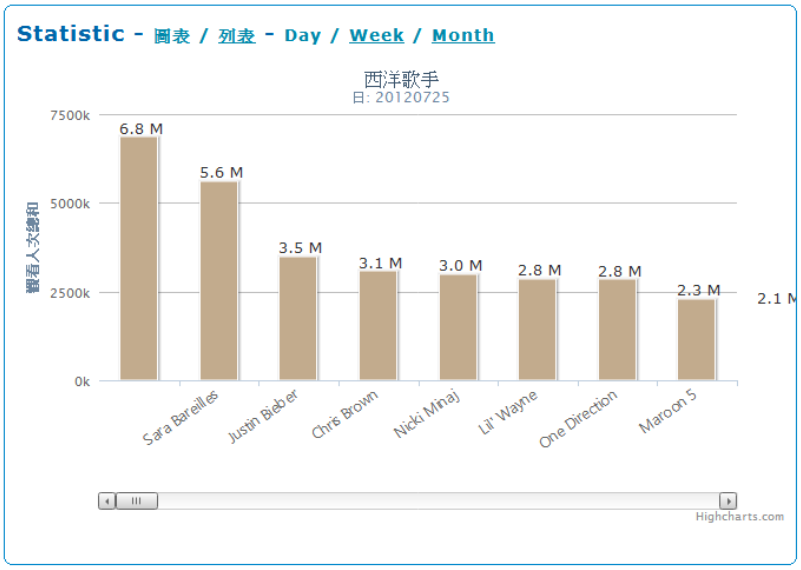
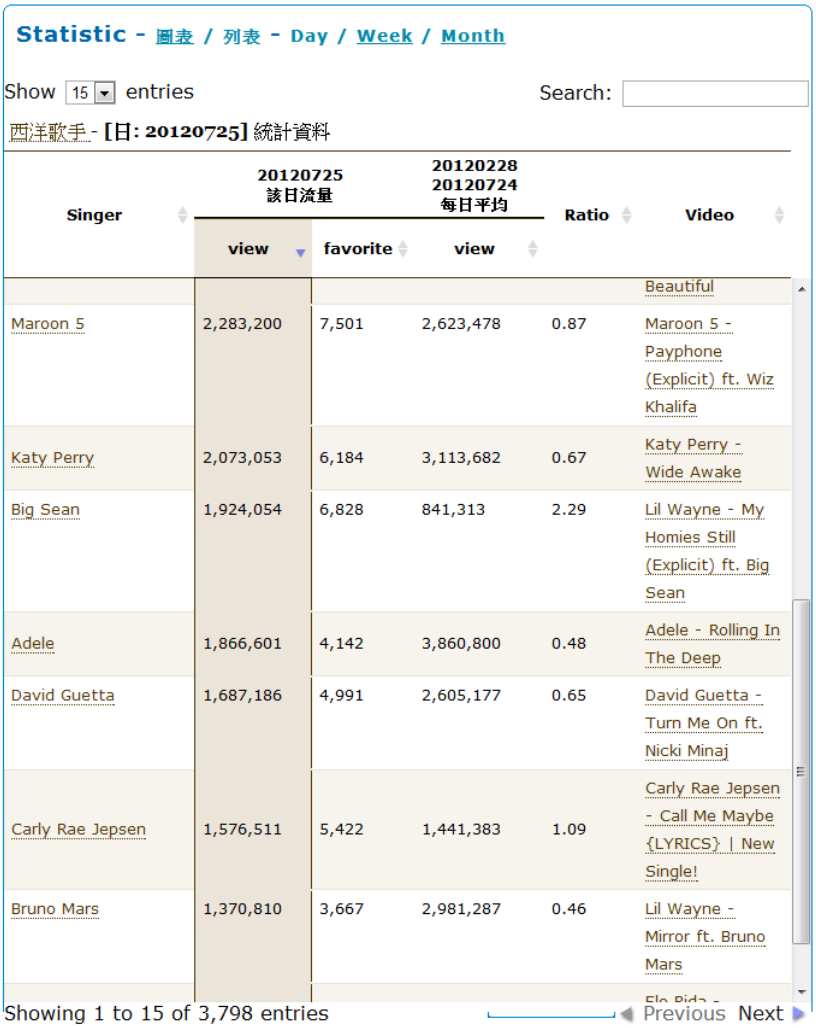


Figure 5. 統計資料圖表呈現



Statistic - 圖表 / 列表 - Day / Week / Month

Show 15 entries Search:

西洋歌手 - [日: 20120725] 統計資料

Singer	20120725 該日流量 view	20120725 該日流量 favorite	20120228 20120724 每日平均 view	Ratio	Video
					Beautiful
Maroon 5	2,283,200	7,501	2,623,478	0.87	Maroon 5 - Payphone (Explicit) ft. Wiz Khalifa
Katy Perry	2,073,053	6,184	3,113,682	0.67	Katy Perry - Wide Awake
Big Sean	1,924,054	6,828	841,313	2.29	Lil Wayne - My Homies Still (Explicit) ft. Big Sean
Adele	1,866,601	4,142	3,860,800	0.48	Adele - Rolling In The Deep
David Guetta	1,687,186	4,991	2,605,177	0.65	David Guetta - Turn Me On ft. Nicki Minaj
Carly Rae Jepsen	1,576,511	5,422	1,441,383	1.09	Carly Rae Jepsen - Call Me Maybe {LYRICS} | New Single!
Bruno Mars	1,370,810	3,667	2,981,287	0.46	Lil Wayne - Mirror ft. Bruno Mars

Showing 1 to 15 of 3,798 entries Previous Next

Figure 6. 統計資料列表呈現

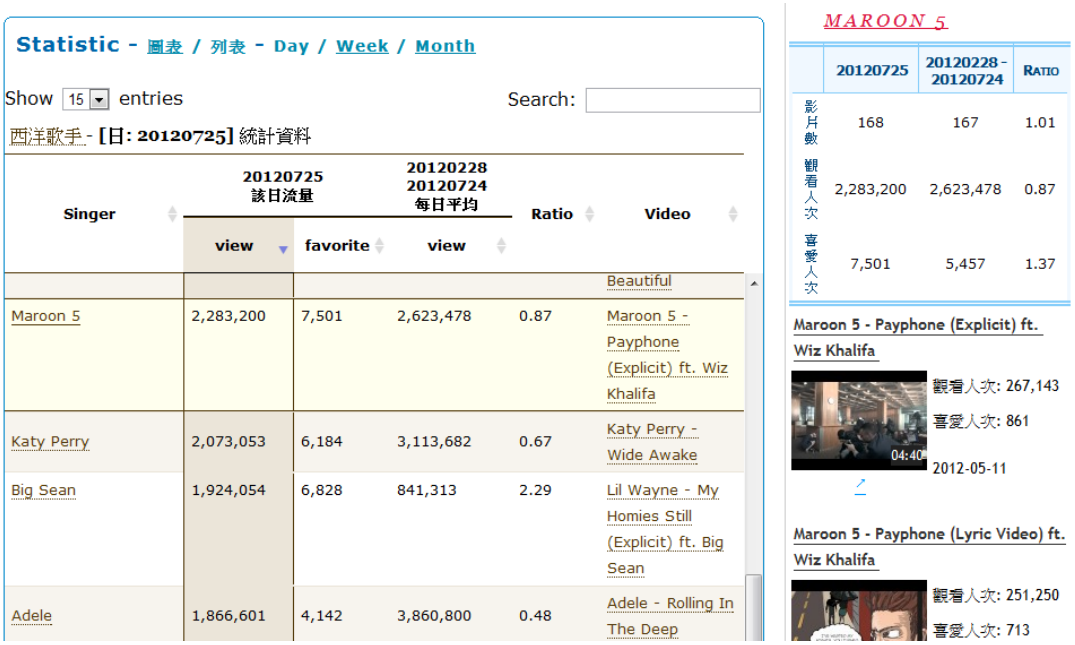


Figure 7. 查看歌手統計資料與觀看次數最多的前五筆影片資料

4.3 User Defined Query

此部分可由使用者自行定義所需統計的詞彙資料。使用者可親自於右方文字方塊內輸入所查之詞彙，亦可直接載入使用者已有之詞彙檔案。左方則提供欲查詢詞彙的類別與統計期間設定，以及詞彙篩選規則條件。(見 Figure 8)

Figure 8. 使用者自行定義詞彙資料

Chapter 5. Conclusion and Future Work

系統目前提供在已知分類類別的 Term DB 條件下，可對影片進行分類瀏覽與詞彙分群，統計該詞彙於各指定期間內的觀看人次與喜愛人次，並與過去期間的觀看/喜愛 次數做比較，查看分詞彙的變化狀況。另也可由使用者自行定義詞彙資料，查看統計數據。

在目前系統提供的分類中，Term DB 在歌詞網取回後，經過人工確認，判斷該詞彙是否是正確屬於該分類，若是原始歌詞網中有分類錯誤的情況，我們也對他進行修正，因此，在 Term DB 資料無誤的情況下，我們已可以透過此系統查看各個流行音樂分類 (華人男歌手、華人女歌手、華人團體歌手、日韓歌手、西洋歌手) 下，依照觀看人次排序所得到的熱門歌手列表。

Multi-Query Match Machine 中，系統利用規則篩選輔佐詞彙比對達到更佳的分類結果，但在規則篩選中，一方面我們達到更高的 precision，但另一方面，也可能造成許多該分類的影片在此過程中被刪除，而大幅度降低 recall 值，因此，如何在提高 precision 值的同時，降低 recall 值減少的幅度，為未來系統的改進方向之一。

統計過程中，我們除了統計近期特定期間內的觀看次數、喜愛次數外，我們也一併對以前的觀看次數、喜愛次數進行統計，一方面想查看各個歌手的觀看次數變化情形，另一方面，我們也想透過此機制得知最近是否有哪位歌手是觀看次數突然增加許多，進而取得最近新出的歌手，或是最近發片歌手的列表。但最後資料呈現時，我們發現在目前機制下產生的此變化數據會有其不確定因素存在，即當我們中途對 Term DB 進行更新，或是我們修改在抓取影片資料時的詞彙內容，可能就會造成相關 Query 的統計資料會由毫無統計數據(0)，突然竄升至幾

千甚至幾萬次的觀看次數增加量，因此在這樣的雜訊干擾下，我們便無法正確取得該分類中，目前竄紅或近期有新活動的歌手名單。

在目前該機制下，須符合的前提條件為：Term DB 已知，目前此部分是由特定網站取得再經由人工確認，因此會對系統造成極大限制，未來此系統著重方向將須針對此部分設計自動化產生機制，減少人力判斷，進而全面提供各種類的影片分類瀏覽。

Chapter 6. References

1. 網路影音服務公司 wistia, http://wistia.com/
2. 線上影音分享網站 YouTube, http://www.youtube.com/
3. Google Doubleclick , http://www.google.com/adplanner/static/top1000/
4. 線上影音分享網站 Youku, http://www.youku.com/
5. 美國市場研究公司 comScore, http://www.comscore.com/
6. 線上影音分享網站 Tudou, http://www.tudou.com/
7. 魔鏡歌詞網, http://mojim.com/
8. YouTube API, https://developers.google.com/YouTube/
9. jQuery 圖表套件, http://www.highcharts.com/
10. jQuery 表格套件, http://datatables.net/
11. Web UI 组件 DHTML, http://dhtmlx.com/docs/products/dhtmlxTabbar/index.shtml