



國立高雄應用科技大學
電子工程系碩士班
碩士論文

應用改良式 K-means 分群法於個人化音樂推薦服務系統
之實現

Application of Refined K-means Clustering to the
Implementation of Personalized Musical Recommendation
System

研 究 生：吳振銘

指導教授：廖斌毅 博士

謝欽旭 博士

中華民國 101 年 1 月

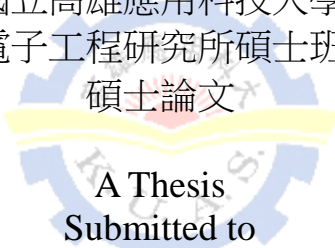
應用改良式 K-means 分群法於個人化音樂推薦服務系統之實現

Application of Refined K-means Clustering to the Implementation of
Personalized Musical Recommendation System

研究生：吳振銘

指導教授：廖斌毅 博士
謝欽旭 博士

國立高雄應用科技大學
電子工程研究所碩士班
碩士論文



A Thesis
Submitted to
Institute of Electronics Engineering
National Kaohsiung University of Applied Sciences
in Partial Fulfillment of the Requirements
for the Degree of
Master of Engineering
in
Electronics Engineering

January 2012

Kaohsiung, Taiwan, Republic of China

中華民國 101 年 1 月

國立高雄應用科技大學研究所學位論文考試審定書

本校 電子工程系碩士班

研究生 吳振銘 所提之論文

應用改良式 k-means 分群法於個人化音樂推薦服務系
統之實現

合於 碩士 資格水準，業經本委員會評審認可。

學位考試委員會

召 集 人

蔡政達

簽章

委 員

席世波
蔡政達

謝鈺旭
林書中

指導教授

席世波 謝鈺旭

簽章

系所主管

楊建明

簽章

中華民國 101 年 1 月 17 日

**Application of Refined K-means Clustering to the
Implementation of Personalized Musical
Recommendation System**

by
Zhen-Ming Wu

A Thesis Submitted to the Graduate Division in Partial
Fulfillment of the Requirements for the Degree of
Master of Science in the Department of Electronics Engineering
National Kaohsiung University of Applied Sciences
Kaohsiung, Taiwan, Republic of China
January 17, 2012

Approved by :

Bing-yih Liao
Yueh-Chung Lin

Chen Da Fai
Lin-Shueh Shieh

Thesis Advisor

: Bing-yih Liao Lin-Shueh Shieh

Department Director :

Kuan-Ming Hung

應用改良式 K-means 分群法於個人化音樂推薦服務系統之實現

學生：吳振銘

指導教授：廖斌毅、謝欽旭 博士

國立高雄應用科技大學電子工程系碩士班

摘 要

近年來全球數位化內容與網際網路的發達，使用網路服務與獲得資訊的人數越來越多；在這資訊爆發時代，使用者接受的資訊範圍也越來越廣，而網路的終端節點不受限於傳統個人電腦，生活周遭的每樣消費電子產品都可經由網路彼此互相溝通、連結，而本篇論文將提出結合音樂推薦系統，與搭載 Android 作業系統的移動式手持裝置，建立起雲端推薦系統伺服器，使用者可以經由個人的移動式手持裝置連結雲端音樂推薦系統伺服器進行匹配運算，最後經由推薦模組產生出推薦結果回傳至移動式手持裝置。本篇論文提出運用推薦系統的分析技術：協同式過濾(Collaborative Filter)模型與內容式過濾模型(Content-Based Filter)，實現基於以音樂為主的推薦系統。收集使用者的對音樂喜好的評分(Rating)建立使用者的使用資訊，並以協同式過濾的概念，找出聆聽興趣、喜好相同的使用者，結合本篇論文提出的改良式 K-means 分群技術，改良傳統 K-means 分群的缺點提高群內的相似度，分群的結果將找出使用者間具有聆聽相同音樂喜好的同群使用者，以喜好類似的同群使用者作為推薦的依據，分析各使用者間的聆聽記錄，計算各使用者聆聽項目間的相關相似性(Correlation)，產生預測的評分結果，雲端推薦系統伺服器將評分高的音樂推薦給使用者。實驗結果將會比較改良式 K-means 分群法與其他分群演算法的結果，以分群的凝聚率(Agglomerate rate)與鑑別率(Discrimination rate)作為分群評估，最後比較使用不同的分群技術對於雲端音樂推薦系統的影響，以 RMSE(Root Mean Square Error)平均誤差值評估不同的分群技術；評估分析對雲端音樂推薦的推薦結果，實驗結果表明：改良式 K-means 分群法對於雲端音樂推薦系統的準確度，有著良好的改善。

關鍵詞：推薦系統、協同式過濾、K-means 分群、相關相似性、凝聚率、鑑別率、RMSE

Application of Refined K-means Clustering to the Implementation of Personalized Musical Recommendation System

Student : Zhen-Ming Wu

Advisors : Dr. Bin-Yih Liao

Dr. Chin-Shiuh Shieh

Department of Electronics Engineering
National Kaohsiung University of Applied Sciences

ABSTRACT

As digitized content and the Internet evolve in recent years, the number of users who obtain information via the Internet keeps growing. In the era of information explosion, the personal computer is no longer the only option as network terminals. A wide range of 3C products can communicate and link together with each other via the Internet. In this study, a music recommendation system based on Android operation system and mobile handhold devices is proposed. The objective is to establish a cloud computing service such that users can get the music matching information via music recommendation model.

Two techniques are involved in our recommendation system: Collaborative Filter Model and Content-Base Filter Model. The rating information is collected by the users' actual rating and the users with similar interest or preference will be found by system via collaborative filter model. An improved K-Means Clustering algorithm is proposed in this thesis which improves the similarity among data in the same group. The clustering classifies users into groups with the same preference. Recommendations are made base on the music preference of users in the same group. The candidates with highest score will be recommended to the users by the cloud recommendation system.

The proposed improvement was experimented against previous approaches. The aggregation rate and discrimination rate were used as the performance indices. And then the Root Mean Square Error was used to evaluate the effectiveness of the recommendation system. Experiment results reveal that the proposed refinement

outperforms previous schemes.

Keyword: Recommendation system, Collaborative Filter, Content-Base Filter, K-means clustering, Correlation, Agglomerate rate, Discrimination rate, RMSE



誌 謝

時光飛逝，很快的碩士兩年的學習生涯已接近尾聲，在求學的過程中無論是研究的方法、問題的解決、以及與業界的交流...等，許多新的嘗試與學習的機會，讓我在攻讀碩士過程中有了新的見解與長進，而在這過程中要感謝一路以來一直陪伴我的父母、教授、學長姐、實驗室的同儕、與學弟妹。

首先，誠摯感謝我的指導教授廖斌毅教授、共同指導教授謝欽旭教授與洪盟峰教授。三位教授在研途上給予詳細與全面性的指導，以及提供許多寶貴研究經驗、資源及機會，使我在研究工作上得以順利進行。感謝廖斌毅教授對於我在碩士的兩年，不斷的接觸產業界新的技術，並給於我們學習的機會，使得我們理論與應用技術能不斷的擴展。感謝謝欽旭教授在我撰寫碩士論文與研究期間給予許多的指導與意見，在研究的過程許多理論與技術上的問題老師都不辭辛勞的協助我，並於百忙之中撥冗與學生討論及解決問題，直到最後的本文撰寫始終在旁協助，對本文悉心審閱與不吝指教，使得研究工作能夠順利進行並有所成果。感謝洪盟峰教授在我碩士的過程給於許多學習的機會，並在日常生活中給予關心、照顧、分享經驗及資源上的支援。同時感謝口試委員蔡政達教授、林聿中教授，給予寶貴的意見與肯定，使得本文能夠更加嚴謹與完整。

感謝博士班的陳怡婷學姊、施宏旗學長在研究過程中，給予協助、解惑與經驗的傳承，使原本底子不足的我，能夠快速的入門與研究。另外，也非常感謝高等資通訊研究室的學長、同儕與學弟妹們，在我研究的過程中遇到挫折替我打氣，有喜悅之事也與我共享，因為有你們共同努力與教學相長，使我的困惑之處可以因此獲得解答。

最後，在此由衷的感謝我的父母親。感謝您們無私地奉獻與無悔的付出，無論是在生活上以及資源上，都呵護備至，雖然讀研究所是辛苦的，但一想到您們讓我沒有煩惱並陪伴我一路升學至此，讓我無後顧之慮的順利完成碩士學位，這其中最大的功勞是屬於您們。在此，謹以本論文表達心中對您們最誠摯的敬意與感謝。

吳振銘 謹誌

中華民國 101 年 1 月

目錄

摘 要.....	i
ABSTRACT.....	ii
誌 謝.....	iv
目 錄.....	v
圖目錄.....	vii
表目錄.....	ix
第一章 緒論.....	1
1.1.前言	1
1.2.研究動機與目的	2
1.3.論文架構	3
第二章 相關研究及文獻探討	5
2.1 推薦系統	5
2.1.1 長尾效應理論(The Long Tail)	7
2.1.2 內容式過濾技術(Content-Based Filtering)	9
2.1.3 協同式過濾技術(Collaborative Filtering)	11
2.1.4 混合式過濾技術(Hybrid Filtering)	14
2.2 分群技術	16
2.2.1 分割式分群演算法(Partitioning Clustering Algorithms)	16
2.2.2 階層式分群演算法(Hierarchical Clustering Algorithms).....	17
2.2.3 基於密度分群演算法(Density-base Clustering Algorithms)	20
2.2.4 基於網格分群演算法(Grid-base Clustering Algorithms)	20
2.2.5 基於模型分群演算法(Model-base Clustering Algorithms)	20
2.3 Android 系統.....	21
2.3.1 JSON 資料交換.....	23
第三章 Android 雲端音樂服務推薦系統建置	24
3.1 音樂推薦服務系統流程與架構	24

3.2 建置 Music Style	25
3.3 建置 User Profile	26
3.3.1 User Profile 的計算	27
3.4 User-base 分群演算法.....	29
3.5 K-means 分群演算法.....	32
3.6 改良式 K-means 分群演算法.....	33
3.6.1 改良初始化群中心.....	33
3.6.2 動態 K 值演算法.....	37
3.7 協同式過濾音樂推薦模組	42
3.8 Android 移動式手持裝置的開發與連結.....	43
第四章 實驗結果與評估	44
4.1 環境建置	44
4.2 分群評估	45
4.4.1 凝聚率(Agglomerate rate)	45
4.4.2 鑑別率(Discrimination rate).....	47
4.3 推薦系統評估	49
4.4 系統展示	50
4.4.1 個人化推薦.....	51
4.4.2 個性化推薦.....	53
第五章 結論與未來展望	56
5.1 研究結論	56
5.2 未來展望	57
參考文獻.....	58

圖目錄

圖 1 Web Sites 統計.....	6
圖 2 長尾效應.....	8
圖 3 降低長尾效應.....	8
圖 4 內容式過濾技術.....	9
圖 5 協同式過濾技術.....	12
圖 6 循序組合.....	14
圖 7 線性組合.....	15
圖 8 階層式分群法.....	17
圖 9 單一鏈結法.....	19
圖 10 完整鏈結法.....	19
圖 11 平均鏈結法.....	20
圖 12 JSON 格式.....	23
圖 13 音樂推薦服務統系統流程.....	24
圖 14 User Profile 建置流程.....	27
圖 15 改良式 K-means 演算法.....	31
圖 16 K-means 演算流程.....	32
圖 17 K-NN 搜尋最遠、最近資料點.....	34
圖 18 一維切割距離.....	35
圖 19 初始化群中心流程.....	37
圖 20 餘弦相似度.....	39
圖 21 餘弦門檻分群.....	41
圖 22 餘弦門檻值.....	41
圖 23 資料交換.....	43
圖 24 Json 解析.....	43
圖 25 凝聚率示意圖.....	46
圖 26 凝聚率.....	47
圖 27 鑑別率示意圖.....	48

圖 28 鑑別率.....	49
圖 29 RMSE.....	50
圖 30 雲端音樂推薦系統首頁.....	51
圖 31 個人化推薦帳號登入.....	52
圖 32 回傳推薦結果.....	52
圖 33 個人化推薦結果.....	53
圖 34 個性化推薦.....	54
圖 35 個人音樂曲風評分.....	54
圖 36 個性化推薦結果.....	55



表目錄

表 1 Android 發行版本	21
表 2 音樂曲風特徵.....	25
表 3 各維度正規化.....	29
表 4 最高次數正規化.....	29
表 5 K-NN 初始群中心.....	36
表 6 三維範例.....	38
表 7 五維範例.....	38
表 8 計算相似度比較.....	40
表 9 資料庫資料集.....	44
表 10 凝聚率.....	46
表 11 鑑別率.....	48



第一章 緒論

在本篇論文的開端，首先介紹研究主題背景，並介紹音樂推薦服務系統的需求，接著探討研究問題並針對問題提出本研究之目的與動機。

1.1. 前言

近年來，隨著網際網路的發達，使用的人口逐年的增加，造成了傳統長度為 32 位元 IPV4 的 IP 位置不足，以致於世界各國全力推廣長度更長的 128 位元 IPV6，IPV6 的 IP 位置為傳統 IPV4 的 2^{96} 倍，屆時網路普及的速度將更為加速，資訊量將比以往更加迅速而大量的增加，資訊爆發的結果所帶來的龐大資訊量，也已遠遠的超過人類所能處理的極限，這時候就需仰賴利用電腦高速的運算處理能力，處理日益膨脹的資訊量，將使用者所需資料篩選出來，而推薦系統 (Recommendation System) 就是因應這問題所發展的其中一種解決方法。

因為網際網路與資訊科技的進步，世界各國皆極力推廣數位化內容，任何各種有形的傳播媒介，皆能進行數位化並經由網際網路傳播至全世界，例如：書籍、影像、音樂等...，而本篇論文將以音樂為例。根據行政院新聞局針對台灣流行音樂產業調查白皮書[13]，內容指出從西元 2000 年開始流行音樂唱片的銷售額逐年下降，反觀受到數位化網路時代的衝擊，數位音樂逐漸的成長並不斷的壓縮到實體唱片的空間，截至目前為止數位音樂已經成為主流。因數位化內容的推動；許多的音樂皆以數位的形式儲存在資訊設備上，累計的數量相當可觀，對使用者來說逐一的試聽是不可能的，為了解決使用者在眾多數位音樂的選擇，並擴展使用者對於新音樂的喜好，本篇論文提出音樂推薦系統 (Music Recommendation System)，建立使用者模型快速的從龐大的數位音樂裡推薦給使用者感興趣的歌曲。

網際網路的發達扮演著現代人獲得新知的方面的重要角色，而音樂推薦系統

是用來提供使用者聆聽會感到興趣音樂的一套服務系統。音樂的發展伴隨著人類的歷史，對人類來說是一種文化的象徵，每個國家都有各自發展的音樂類型、與曲風，累計了數量可觀的音樂量，現今網路技術的發達，使用者聆聽音樂的類型已經無國界之分，聆聽的音樂數量已遠遠超出人們花上一輩子的時間都無法聆聽完的音樂量，音樂推薦系統就是為了解決使用者無法處理的龐大資訊量，利用電腦強大的演算能力，將使用者的聆聽紀錄與使用者的聆聽興趣作為輸入電腦的推薦數據，分析使用者的聆聽紀錄與使用者的聆聽興趣，找出與使用者興趣相符的音樂項目，結果將會推薦出使用者感到興趣的音樂項目提供給使用者作為聆聽的選擇。

音樂對於人們來說是一個能夠適時適地的休閒，也是一種可以放鬆、調整個人心情的旋律，不論是在工作的場合播放著能夠使心情愉悅增進工作效能的音樂、或是在休息睡眠前播著能讓人放鬆、幫助入眠的輕鬆曲風，音樂對於人類的生活文化有著極深的影響。憑藉著現代科技的快速進步，移動式手持裝置也不再是只有傳統的電話功能，而是包含著許多的附加價值與應用，網路便是移動式手持裝置上一個成功的附加價值，藉由讓移動式手持裝置上網的功能，使得使用者跳脫只能在個人電腦前上網的框架，憑藉著移動式手持裝置就能連結網路窺得全世界。隨著硬體開發與製程技術的進步，移動式手持裝置的硬體能力也越來越強大，多媒體應用便是移動式手持裝置另一個成功的附加價值，結合網路與多媒體的移動式手持裝置，就像一部小型的移動電腦一樣，走到哪都可以隨時聆聽音樂、觀賞影片，但是隨之而來的龐大資訊量還是無法得到有效率的運用，人類一天的時間有限，所能處理的資訊量也有限，這時就必須仰賴電腦的高速演算能力，幫助人類找出有用的資訊，作為提供人們的一種參考，減少因過濾龐大的資訊量所耗損的時間與人力成本。

1.2. 研究動機與目的

推薦系統的起因是因為網路的普及化與應用發展，造就了今日資訊爆發的數位科技時代，資訊處理幫助使用者將過多的資訊進行化繁為簡的動作，而推薦系統的成形是由 Resnick 與 Varian 在 1997 年所共同正式提出，在還沒正式提出前，

推薦系統的概念就已發展數年，而推薦系統的常用的相關技術，主要是以內容式過濾[1][6]與協同式過濾[1][6]為主。內容式過濾是以項目的特徵為主，分析使用者聆聽過的音樂特徵，再比對音樂資料庫裡具有類似特徵的音樂推薦給使用者，這樣的推薦方式好處是：準確度高、符合使用者的聆聽興趣，但缺點是：推薦結果將會侷限於使用者所聆聽過的音樂特徵，這種結果顯然是不樂見的，爾後為了增加系統的拓展性，發展了協同式過濾。協同式過濾的概念，比較像是我們口中的口碑效應，譬如：有一天某 A 對某 B 說：某某歌滿好聽的。這樣的作法就是一種推薦。找出跟自己喜好相近的使用者，再與自己相近的使用者進行推薦，這就是協同式過濾的方法，而本篇論文以協同式過濾為基礎，利用改良式 K-means 分群技術，找出跟使用者有著相同喜好的其他使用者，再以同群的其他使用者為推薦的基礎，進行使用者與音樂項目間的相關相似性計算，選出相似性較高的音樂項目，產生預測的評分，將評分高的音樂推薦給使用者。

對人們來說音樂具有適時適地的性質，結合多媒體移動式手持裝置無論何時何地都能聆聽音樂，為了能更有效地提供使用者聆聽音樂的選擇，本篇論文結合搭載 Android 系統的多媒體移動式手持裝置與雲端推薦系統伺服器連結，在 Android 系統上設計一套連結雲端推薦系統伺服器的 GUI(Graphical User Interface) 介面，並在雲端推薦系統伺服器建立起個人化音樂聆聽評分紀錄，藉由雲端推薦系統分析使用者的聆聽評分紀錄並經由推薦模組的演算，將推薦結果回傳至移動式手持裝置呈現再螢幕上，使用者可透過移動式手持裝置上的觸碰面板選擇自己感興趣的音樂，結合了移動式手持裝置、網路、雲端推薦系統伺服器，使得使用者在聆聽音樂上有更多的選擇，利用雲端推薦系統伺服器增加使用者聆聽更多可能會感興趣的音樂，並結合移動式手持裝置讓使用者可以更適時適地的聆聽音樂，而不是侷限在個人電腦的框架上。本篇論文的目標在於結合移動式手持裝置、網路、與雲端推薦系統伺服器，將這些要素整合起來，實現具有智慧的音樂推薦服務隨身聽的目標。

1.3. 論文架構

本篇論文共分成五章，第一章探討研究背景與音樂推薦服務系統的需求，第二章將提到為何需要推薦系統；而推薦系統將會對市場帶來的效益，並探討推薦系統的發展與推薦系統相關技術，結合 Android 系統對推薦系統的擴展性。第三章將針對本篇論文提出的改良式 K-means 分群技術進行分析，比較傳統 K-means 演算方法與缺點，並說明建置雲端音樂推薦系統的步驟與連結移動式手持裝置的方法。第四章比較改良式 K-means 分群技術與其他的分群技術的分群結果，評估群集的凝聚率與鑑別率，並比較使用不同的分群技術對推薦系統的影響，以 RMSE 評估方法比較推薦的結果。第五章結論與未來展望，對先前的研究過程與法作個總結，並針對本研究提出未來的發展與改良。



第二章 相關研究及文獻探討

本章節欲探討推薦系統相關技術，並說明為何需要推薦系統以及推薦系統是運用何種資料過濾技術，如何分析使用者的個人情報，並推薦出使用者感興趣的結果，本篇論文將會與搭載 Android 系統的移動式手持裝置結合，提供適時適地的音樂推薦系統服務。首先，在 2.1 節中會介紹推薦系統的技術發展，與推薦系統常用的過濾技術，協同式過濾技術(Collaborative Filter)、內容式過濾技術(Content-Base Filter)、與混合式過濾技術(Hybrid Filter)；接著 2.2 節中會介紹現有的分群技術，探討目前已知的分群方法，並說明不同的分群的差異；最後在 2.3 節中會介紹移動式手持裝置系統：Android，並說明目前在 Android 上的需求與如何使用網路與 Server 連結進行應用開發。

2.1 推薦系統

在網路發達的現代，充斥著大量的資訊，這些大量的資訊並不單單只靠人力就能完全掌握，搜尋引擎就是解決充斥著大量的資訊，卻又無法快速地找出使用者需求的一個最好的例子；人們經由網際網路瀏覽網頁，但是全世界的網站琳瑯滿目，根據 Netcraft 到 2011 年 6 月份的統計[8]，全世界的 Web Sites 的總數目前已達到 346,004,403 個，如圖 1 所示。

目前全世界網站的總數高達 3 億多組，人們若要尋找資訊只靠人力是無法完成的，這時就必須仰賴搜尋引擎，由使用者輸入關鍵字，提供搜尋引擎服務的廠商就會利用他們電腦硬體的強大演算能力，幫您從 3 億多組網站已不到 1 秒鐘的時間，找出使用者所需要資訊，這就是一種資訊過濾技術(Information Filtering)。而推薦系統就是其中的一種，為了避免使用者在搜尋資訊過程中消耗掉太多無謂的成本，而提出的一種資訊過濾機制，推薦的依據以使用者對於項目的喜好、興趣、或行為等...，透過分群(Cluster)或機器學習(Machine Learning)等...分析技術，分析內容或藉由分析其他相似使用者的經驗，來對使用者作出一些建議與推薦，

推薦出使用者可能有所需求的潛在資訊、服務或產品。

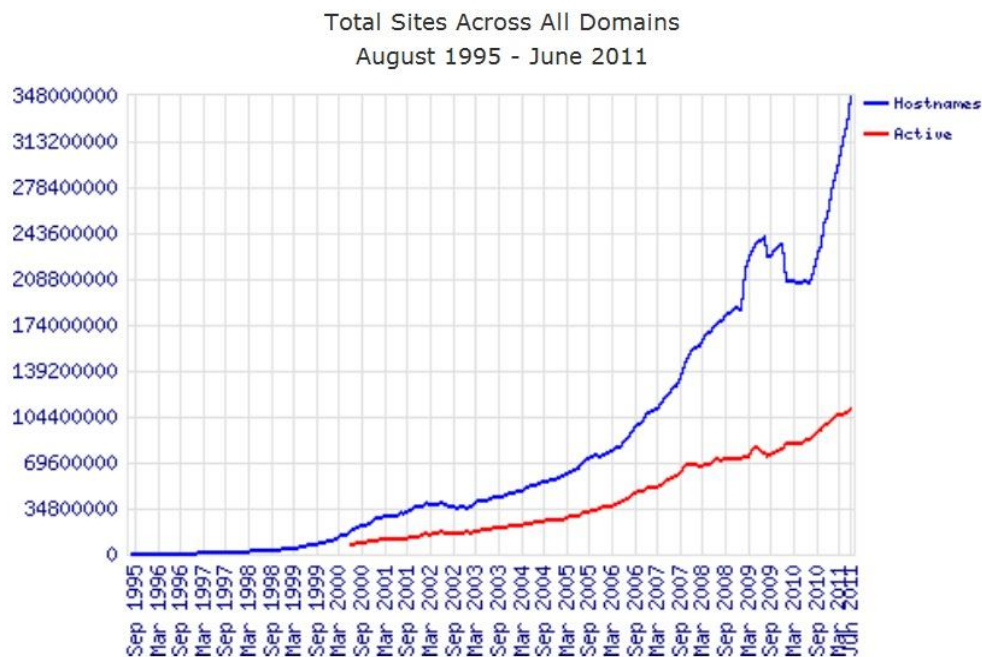


圖 1 Web Sites 統計

(資料來源：<http://news.netcraft.com/>)

服務商可藉由取得使用者的使用紀錄，分析判斷使用者的喜好行為作為推薦預測的依據，在消費行為上達到刺激使用者進行消費，以增加商品銷售的機會。依據推薦資訊的方式，將推薦方式分為四類[15]：

1. 非個人化(Non-Personalized)

此種推薦方式是基於依產品的熱銷程度推薦給使用者，此類推薦不針對個人化使用者提供個別資訊或產品，推薦結果只將熱門產品或資訊全部呈現給使用者，因此每位使用者將會看到相同的產品或資訊。

2. 基於屬性(Attribute-Based)

此種推薦方式是基於使用者對資訊或產品特定屬性的喜好，提供符合該項屬性的資訊或產品。

3. 項目相關(Item-to-Item Correlation)

此種推薦方式是基於利用過去的歷史資料找出常被同時購買的項目，意即找出項目之間的關聯法則，再利用關聯法則去作項目之間的推薦。

4. 使用者相關(People-to-People Correlation)

此種推薦方式會針對目標使用者找出具有相似喜好的使用者，再根據這些相似喜好的使用者對產品或資訊的評價找出適當的項目推薦給使用者。

2.1.1 長尾效應理論(The Long Tail)

長尾效應一詞最初由《連線雜誌(WIRED)》的雜誌總編輯 Chirs Andersony 在 2004 出版的《連線雜誌(WIRED)》[9][10]上發表的，用來描述如 Amazon、Netflix、eBay、iTunes、Rhapsody 等...，以網站作為經營、銷售模式的商業服務網站。內容指出在銷售網站上的商品，是指那些原來不受到重視的銷量小、種類多的產品或服務，但由於總數量巨大，累積起來的總收益超過主流產品的現象。在網際網路領域，長尾效應尤為顯著，這是因為使用者的瀏覽視野變寬了，所造成的現象就是主流商品一樣熱賣，而非主流商品則因為網際網路的便利增加了與消費者接觸的機會，進而提升買氣。

如圖 2 所示，可以看到縱軸為銷售量；橫軸為商品數量，將商品的數量與商品的銷售量統計出來，可繪出一條「長尾」曲線，對於這種現象，在商業界普遍對於商品的數量與銷售量存在著所謂的「80/20 法則」，該法則的意義是指業界 80%的業績來自 20%的產品，以此種看法來表示圖 2，是指商業經營看重的是銷售曲線左端的少數暢銷商品也就是頭端，曲線右端的多數冷門商品也就是尾端，被該定律定義為不具銷售力且無法獲利的區塊，換句話說在所有商品總數裡只有銷售量高的 20%，所帶來的利益會占總收益的 80%，由於網際網路的崛起以及資訊過濾的進步，Z. Zaier 在這裡提到推薦系統與長尾理論的關係[11]，因為推薦

系統的影響，廣泛的增加商品銷售層面，減少商品的滯銷率，讓剩下 80% 的產品都有機會銷售，降低長尾效應如圖 3 所示，而這些具有長尾特性商品將具有增長企業營利空間的價值。不僅如此，佔商品總數 80% 的長尾商品的規模大得驚人，成功的開拓長尾商品的價值，其商品的營業總值甚至可與暢銷商品抗衡。

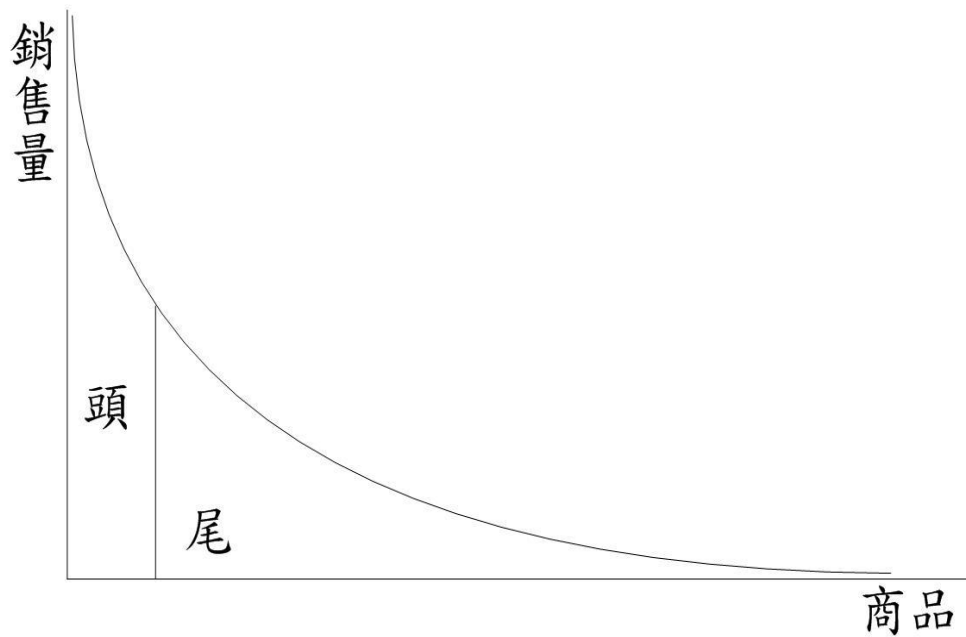


圖 2 長尾效應

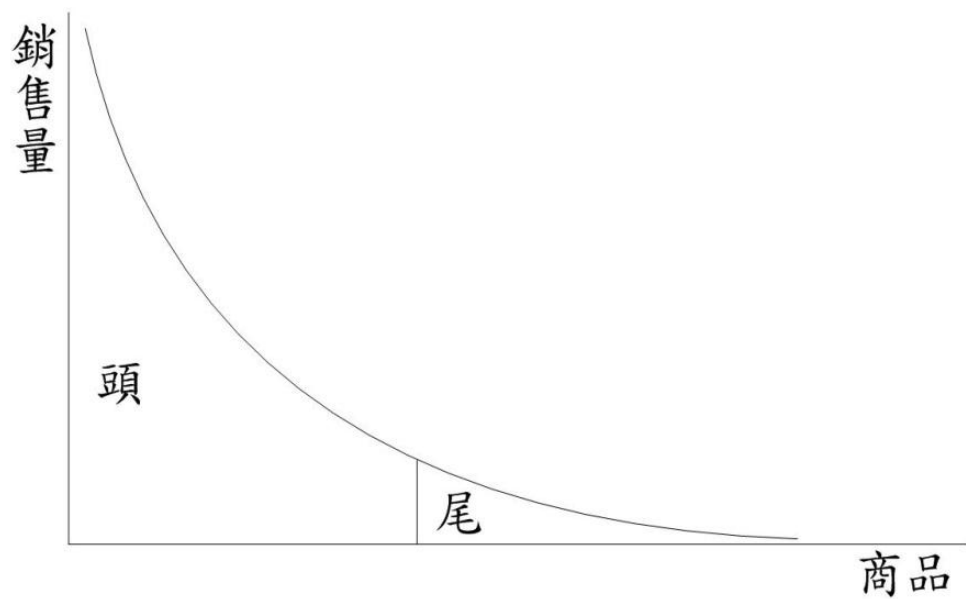


圖 3 降低長尾效應

2.1.2 內容式過濾技術(Content-Base Filtering)

內容式過濾技術又稱為以特徵為基礎之過濾(Feature-Based Filtering) 或是認知過濾(Cognitive Filtering)，它的概念是在於一個人面對不同的選擇做出決定時，會優先選擇印象中接近或者是相似的關係，所以主要的決定在於人對物品的分析，而不是在於評價，所以系統會針對該使用者對物品的喜愛程度，再將數值交由推薦預測模組，算出該使用者可能會感興趣的特徵，進而找出該使用者可能喜歡的物品。

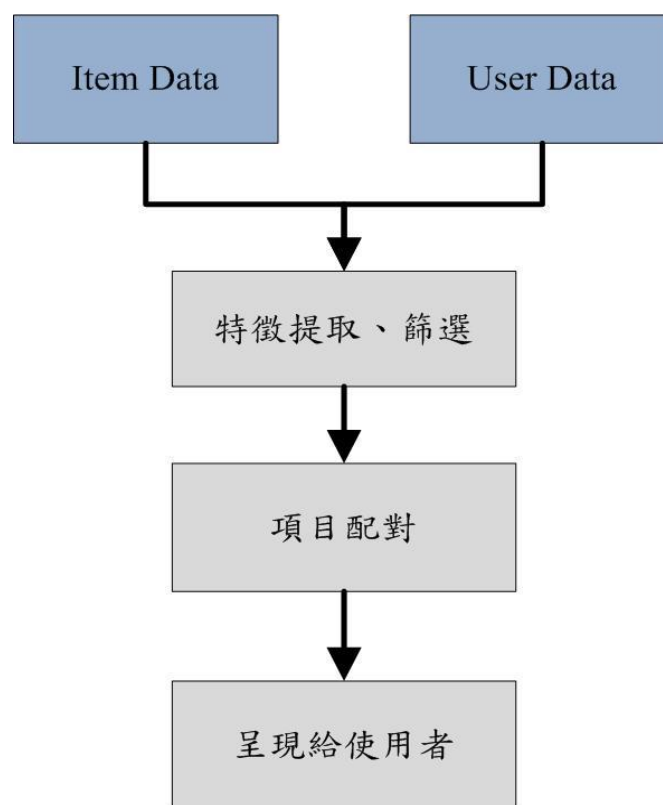


圖 4 內容式過濾技術

圖 4 就是一個簡單的以內容式過濾技術為基礎的推薦系統，它的運作方法為系統先收集使用者的所有資訊。有些研究學者將收集使用者資訊的行為稱為以特徵為基礎(Feature-Based)，其概念做法是源自於資訊擷取(Information Retrieval)的領域所延伸而來的，主要的作法是分析項目的屬性特徵，並以項目的內容或屬性為每個物品建立向量，再以餘弦公式(Cosine)如公式 2.1，判斷兩個物品間的向量夾角 θ 值， θ 值越小代表的是兩個物品間的相關相似度(Correlation Similarity)越

高，反之則越小。

$$\cos \theta = \frac{\vec{a} \cdot \vec{b}}{|\vec{a}| |\vec{b}|} = \frac{a_1 b_1 + a_2 b_2}{\sqrt{a_1^2 + a_2^2} \sqrt{b_1^2 + b_2^2}} \quad (2.1)$$

內容式過濾技術主要是依據使用者個人過去所喜愛的或接觸過的物品的使用紀錄，得知使用者的興趣喜好，並加以分析，進而推薦與使用者喜好相近的物品。作法是根據使用者在過去對於物品喜好或使用的紀錄中，找出哪些物品的屬性特徵有類似或雷同的，將符合的物品推薦給使用者，以本篇論文的音樂推薦系統為例，某位使用者在過去的使用紀錄中幾乎都是聆聽國內知名的某位男性歌手的歌曲，系統便會認為該使用者對於該知名男性歌手的歌曲有著很大的喜好，所以當使用者下一次在使用推薦系統時，該名男性歌手如果推出新的歌曲，或是其他沒聆聽過的歌曲，音樂推薦系統便會立刻將該名男性歌手的最新歌曲，或是其他沒聆聽過的歌曲推薦給使用者，所以以內容式過濾技術為基礎的推薦系統，首先要做的是收集並了解使用者的喜好。

內容式過濾推薦系統隨著使用者的使用資料越多，針對個人的推薦結果就越準，但是 U. Shardanand[12]針對以內容式過濾技術為主的推薦系統，提出部份的缺點：

1. 有限的分析(Limited Analysis)：

分析物品內容只能用一個特徵集合來表示，但卻無法對多媒體屬性進行分析，例如：聲音、圖片、藝術品、影像，因為這些屬性的特徵並無明確的特徵，所以沒有辦法利用一個特徵集合來表示。

2. 過度特殊化(Over Specialization)：

呈現給使用者的推薦結果僅能接收到與過去相類似的推薦項目，而無法找出

與過去喜好有所不同，且具意義性的潛在性推薦，例如：使用者對於某位歌手的歌曲有著很大的喜好，所以當使用者下一次在使用音樂推薦系統時，音樂推薦系統將會推薦該名歌手的歌曲給使用者，但是這樣造成的結果是系統內記錄著許多使用者對於該名歌手的聆聽紀錄，下一次使用音樂推薦系統時，系統也只會推薦該名歌手的歌曲，而無法找出與過去喜好有所不同的推薦。

3. 同義異名(Synonymy)：

是指物品擁有不同的名稱，事實上卻是指同一件東西，譬如：以牛肉麵為例，市面上牛肉麵種類多，命名的方式也五花八門，但是其本質就是牛肉麵，類似像同義異名的物品也不少，而利用內容式過濾的方法並沒有辦法做出有效的分辨。

4. 新使用者(New User)：

如果在推薦系統上完全不存在著該名使用者的使用記錄時，會導致推薦系統無法掌握該名使用者的喜好特徵，會導致無法很正確且即時的做出有效的推薦結果。

解決的方法：針對第一、三點，可以運用機器學習(Machine Learning)的方法找出適合代表目前狀態的特徵。針對第二點；可以考慮使用協同式過濾技術、或者混合式過濾技術拓展使用者對於新項目的喜好。針對第四點，可以在使用者第一次登入系統的時候，運用註冊系統或者是詢問的方式，詢問使用者有哪些喜好，做為系統初始推薦的依據。內容式過濾推薦系統是比較使用者的喜好與資料內容間的關連性，再進行資料上的推薦。

2.1.3 協同式過濾技術(Collaborative Filtering)

協同式過濾技術可以算是推薦系統的始祖，協同式過濾技術在 1992 年由 Goldberg 提出，但當時協同式過濾技術是被用來過濾大量電子郵件，後來便將系

統延伸到所有的電子文件，之後誕生了一套叫做 Tapestry 的系統，而這套系統就是協同式過濾技術的始祖，使用者可以透過 TQL(Tapestry Query Language)的語法建立查詢(Query)的規則，使用者可以設定不同的查詢規則過濾出自己有興趣的信件，但此系統並非是主動的，因為結果並不是由系統自動產生的，使用者仍需輸入查詢規則的方式來過濾使用者喜好的文件。

1994 年 Resnick 在 GroupLens 系統中加入了相似的概念，採用 Neighborhood-Based 的演算法，再以 Pearson 相關係數計算每位使用者的相似度，以找尋合適的鄰居用戶(User-Neighborhood)，進而對使用者可能感興趣的內容進行推薦，使用者就無需再手動的建立查詢的方法來過濾資訊。

接著 Tapestry 與 GroupLens 之後，爾後的系統 Ringo (Shardanand and Maes, 1995)與 Bellcore Video Recommender (Hill et al., 1995)，提供使用者一個排序過的商品推薦清單(ranked list)，該方法為推薦系統發展至今的主要核心工作之一，而推薦系統的發展更趨於成熟，並廣泛的應用於各式各樣的商務系統，例如 Amazon、eBay...等，都是在電子商務網站中使用推薦系統成功的例子，協同式過濾推薦系統嘗試學習每位使用者的興趣與嗜好並儲存起來，讓每位使用者在瀏覽商品時提供個人化的推薦服務，推薦使用者可能感興趣的商品資訊，使用者不需要花費許多的時間找尋其需要的商品。

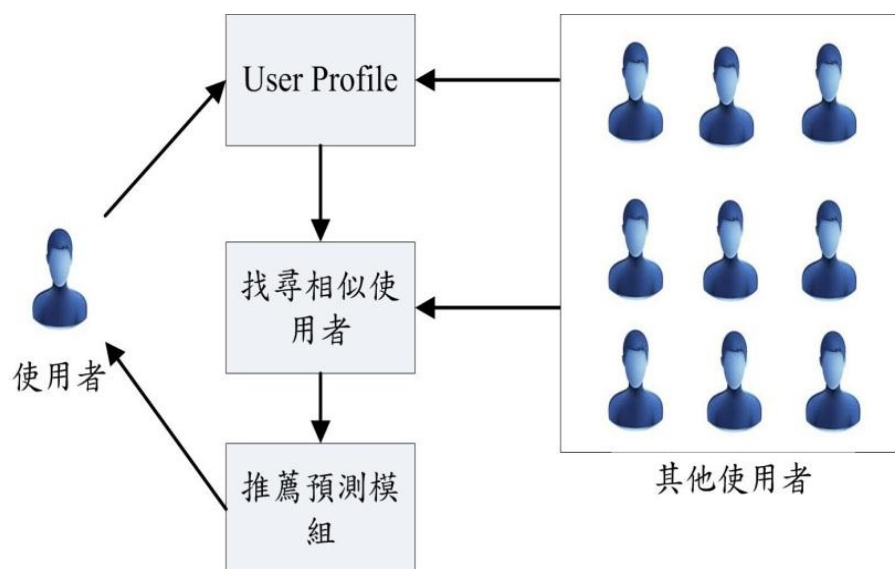


圖 5 協同式過濾技術

圖 5 是一個協同式過濾推薦系統的推薦流程，推薦的步驟如下：

1. 建置一個使用者與其他使用者的使用者輪廓(**User Profile**)，其內容包括使用者對於物品或者是特定資訊的評分(**Rating**)或評價(**Evaluation**)，透過使用者的評分或評價，取得使用者對於物品或特定資訊的喜好。
2. 比較使用者與其他使用者的使用者輪廓，並進行相似度計算；計算出使用者與其他使用者之間的相似度。
3. 找出鄰近的使用者，建構出一個群組，該群組是與使用者相近的其他使用者，再以群組內的使用作為推薦的基礎
4. 使用鄰近使用者的群組進行推薦，加權群組內鄰近使用者的使用者輪廓，作為對新的項目評價，評價高的推薦給使用者。

雖然協同式過濾推薦系統是目前最普遍，也是最常使用的方法之一，但還是存在著一些使用上的缺點：

1. 新使用者(**New User**)：對於協同式過濾推薦系統來說，新的使用者不存在著該名使用者的使用記錄，以至於無法產生使用者輪廓，在進行相似度運算時便無法判斷，該名使用者是否與使用者的相似度高，會導致推薦系統無法掌握該名使用者的喜好特徵，會導致無法很正確且即時的做出有效的推薦結果。
2. 稀疏化(**Sparsity**)：協同式過濾推薦系統通常都會利用對於相當龐大的產品或物件作為評價與推薦，但是使用者所接觸到的產品或物件大部分都只佔全部數量非常小的比例。當產品或物件的數量大於使用者，或兩者的比例差距十

分懸殊時，使用者對於產品或物件的評價，將會佔非常小的比例，產生的推薦結果將會有限。

3. 同義異名(Synonymy):是指物品擁有不同的名稱,事實上卻是指同一件東西,同義異名問題在內容式過濾技術與協同式過濾技術都存在著此類的問題,而類似同義異名的物品也不少,利用協同式過濾的方法沒有辦法做出有效的分辨。

其中冷啟動問題(Cold Start)[6]是從第 2 項延伸出來的,當一樣新的商品加入了推薦系統,此時並未有任用的使用者使用過或對此商品做評價,推薦系統自然是無法找到推薦者來推薦,所以有許多學者開始針對這個問題去修正協同式推薦系統,進而做出更高效能的推薦系統。

2.1.4 混合式過濾技術(Hybrid Filtering)

混合式過濾推薦系統,是結合內容式過濾推薦系統與協同式過濾推薦系統的方法,調整初始資料或新使用者問題,Q. Li[6]針對混合式過濾推薦系統提出了兩種組合模型:循序組合(Sequential combination)圖 6 所示、線性組合(Linear combination)圖 7 所示。

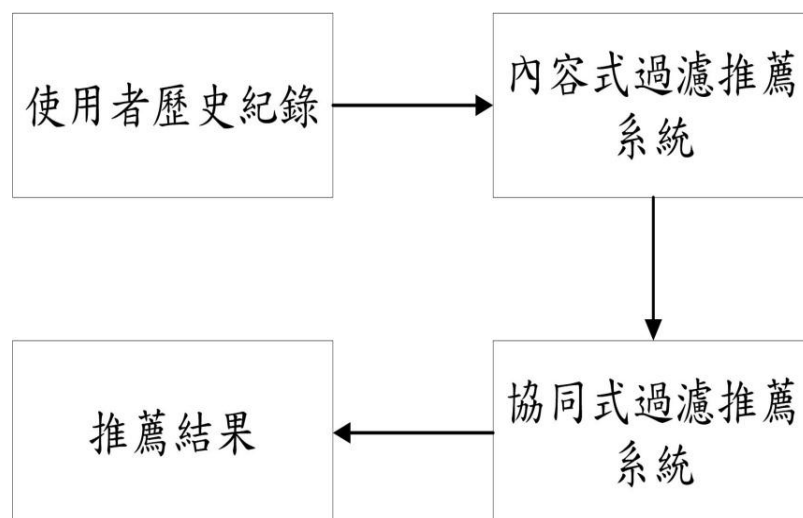


圖 6 循序組合

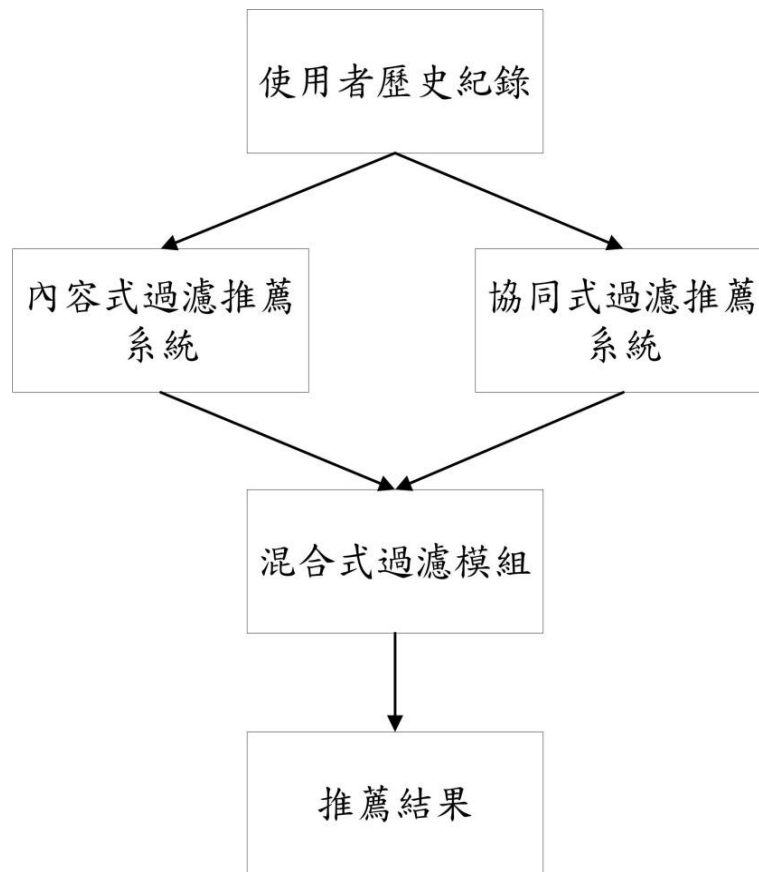


圖 7 線性組合

圖 6 循序組合推薦系統的作法分為兩步驟：

1. 先使用內容式過濾技術，找出使用者的興趣或喜好，再將獲得的資訊交由協同式過濾技術分析。
2. 協同式過濾技術收到使用者的興趣或喜好，找出相似的使用者將結果呈現推薦給使用者。

圖 7 線性組合推薦系統的作法是將使用者資訊由內容式過濾技術與協同式過濾技術共同分析，最後將分析出的結果交給混合式過濾模組依照權重比例將結果呈現推薦給使用者。

混合式過濾推薦系統只是一種概念，系統可依使用者對於物品屬性，喜好，

資料的內容或資訊的特徵...等，做出不同的組合方式與調整，組合的方式也不限於內容式過濾技術或協同式過濾技術，也可以結合類神經網路、統計模型、分群技術...等過濾技術，建構出對使用者更具有參考價值的推薦系統。

2.2 分群技術

在資料探勘(Data mining)這一門學科領域中，分群技術(Clustering technology)是一項常用的資訊分析技術，在資訊充斥的時代，大量的資料帶來了大量的知識，但在龐大的資料中，使用者想找出彼此間有關連性的資料，是非常不容易的；逐一的搜尋比對，往往會花費不少的成本與人力，為了減少耗損的時間與成本，運用分群技術將龐大的資料一一區別；分群技術的概念是將全部的資料依照演算公式中所設定的條件分成許多資料群集，分群的結果具有同一群的資料，彼此間相似度高，群跟群之間的相似度低之特性，使用者再依所需的目的是分析各群集所代表的意義。目前常見的分群技術有：(1) 將資料依使用者需求劃分成 K 群的分割式分群演算法(Partitioning Clustering Algorithms)、(2)將資料一層層的逐漸聚合(Agglomerative)或分裂(Divisive)的階層式分群演算法(Hierarchical Clustering Algorithms)、(3)找出資料中彼此間某一區域密度高於門檻值的基於密度分群演算法(Density-base Clustering Algorithms)、(4)將資料空間切隔成許多網格的基於網格分群演算法(Grid-base Clustering Algorithms)、(5)假設每一組群集皆有一個符合的數學模型的基於模型分群演算法(Model-base Clustering Algorithms)。

2.2.1 分割式分群演算法(Partitioning Clustering Algorithms)

在資料分析的技術裡，分割式分群演算法是發展較早的分群演算法，使用者事先須針對演算法指定欲分群的群數 K ， K 的值不能大於分群的資料數 N ，在分群開始前會先產生出一組初始群中心，初始群中心的決定方式有很多種；最普遍的做法都是以亂數選出初始的群中心，接著開始評估各資料點與群中心之間的關係，常用的評估方式是衡量兩者之間的距離，距離的計算方法有：Euclidean distance(公式 2.2)、Manhattan distance(公式 2.3)、Minkowshi distance(公式 2.4)等...

$$d(A, B) = \sqrt{\sum_{i=1}^n (A_i - B_i)^2} \quad (2.2)$$

$$d(A, B) = \sum_{i=1}^n |A_i - B_i| \quad (2.3)$$

$$d(A, B) = (\sum_{i=1}^n (A_i - B_i)^m)^{\frac{1}{m}} \quad (2.4)$$

代表的分群法有：K-means[2]、PAM[33]、CLARA[33]、CLARANS[21]。

2.2.2 階層式分群演算法(Hierarchical Clustering Algorithms)

階層式分群演算法並不是一次性的分群方式，而是利用凝聚法(Agglomerative)或分裂法(Divisive)一層層分出，直到使用者指定的群數為止，凝聚法是由多至少(Bottom-up)作法是一開每一筆資料皆視為一群，將相似的資料逐漸的聚集起來，而分裂法恰好相反，他是使用由少至多(Top-down)方式；作法是一開始所有的資料視為一群，將不相似的資料逐漸分裂出來，階層式分群法分群過程如圖 8 所示：

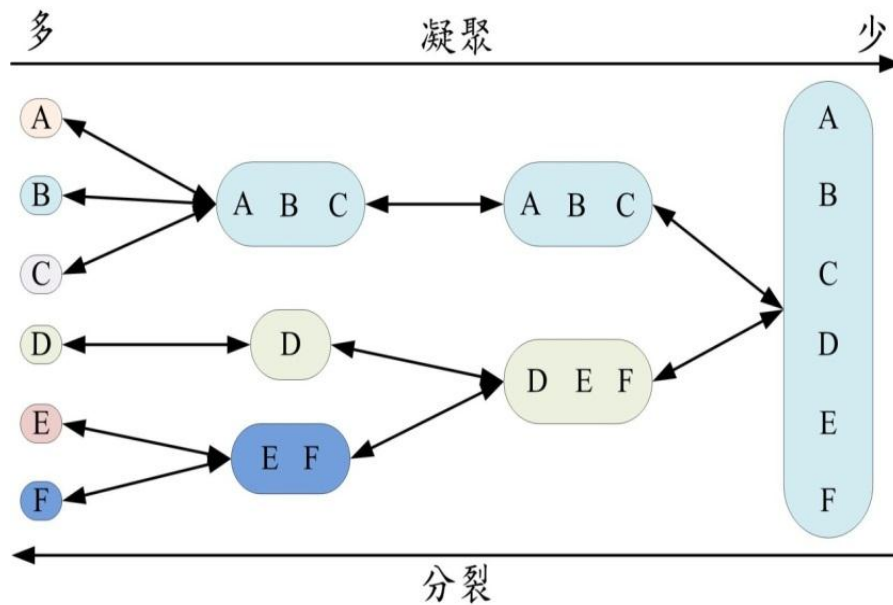


圖 8 階層式分群法

代表的階層式分群演算法有：CURE[24]、ROCK[25]、CHAMELEON[26]、AMOEBA[27]、BIRCH[28]。

2.2.2.1 CURE 演算法

本篇論文提出改良式 K-means 分群演算法做為改良初始化群中心而造成分群不佳的問題，將會引用 B. Chen[16][29]等其他學者提出結合階層式分群演算法作為初始化群中心的概念，本篇論文將引用 CURE 演算法[24]作為選出初始群中心的方法，並與其他方法做比較。

CURE 演算法是凝聚型的演算法，其作法是一開始每一筆資料點視為各自獨立的一群開始聚合，流程如下：

- 一、使用者決定欲分群數 K ， $K \leq I, I \in \text{All data}$ 。
- 二、隨機選擇 K 組群集作為聚合的群集。
- 三、使用聚合演算法找出相似的群集。
- 四、將相似的群集合併產生新的群集
- 五、反覆迭帶直至全部群數剩 K 群。

其中凝聚型演算法如何找出相似的群集，最關鍵的就是聚合演算法，常見的聚合演算法有：單一鏈結聚合演算法(Single-link agglomerative algorithms)、完整鏈結演算法(Complete-link agglomerative algorithms)、平均鏈結演算法(Average-link agglomerative algorithms)：

2.2.2.1.1 單一鏈結聚合演算法

單一鏈結法又稱為最小鏈結法，是以兩組群集中距離最短的點如圖 9 所示，判斷兩群的相似性，取距離最小的群合併(公式 2.5)。

$$D_{\min}(A_u, B_v) = \min_{p \in u, p' \in v} |p - p'| \quad (2.5)$$

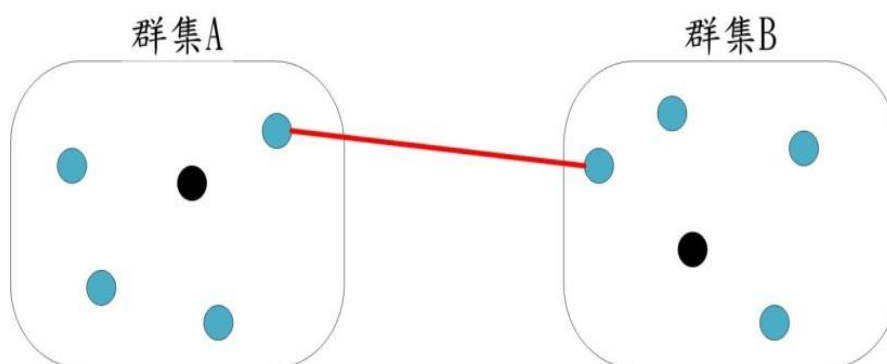


圖 9 單一鏈結法

2.2.2.1.2 完整鏈結聚合演算法

完整鏈結法又稱為最遠鏈結法，是以兩組群集中距離最遠的點如圖 10 所示，判斷兩群的相似性，取距離最小的群合併(公式 2.6)。

$$D_{\max}(A_u, B_v) = \max_{p \in u, p' \in v} |p - p'| \quad (2.6)$$

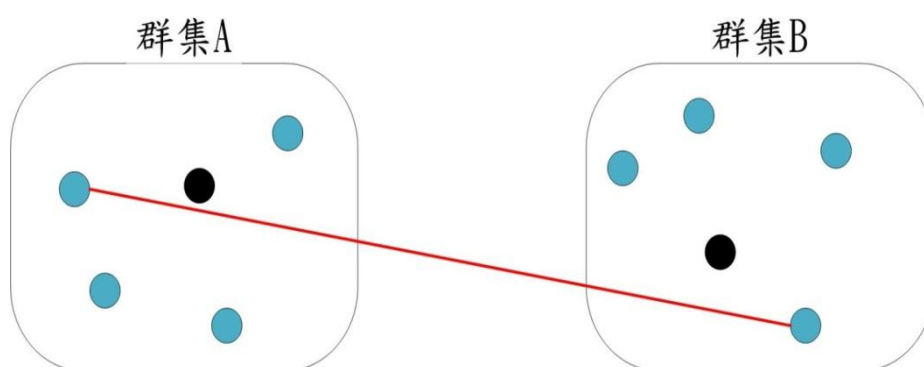


圖 10 完整鏈結法

2.2.2.1.3 平均鏈結聚合演算法

平均鏈結法是以兩組群集各自全部資料的平均距離如圖 11 所示，判斷兩群的相似性，取距離最小的群合併(公式 2.7)。

$$D_{max}(A_u, B_v) = \frac{1}{n_u n_v} \sum_{p \in u} \sum_{p' \in v} |p - p'| \quad (2.7)$$

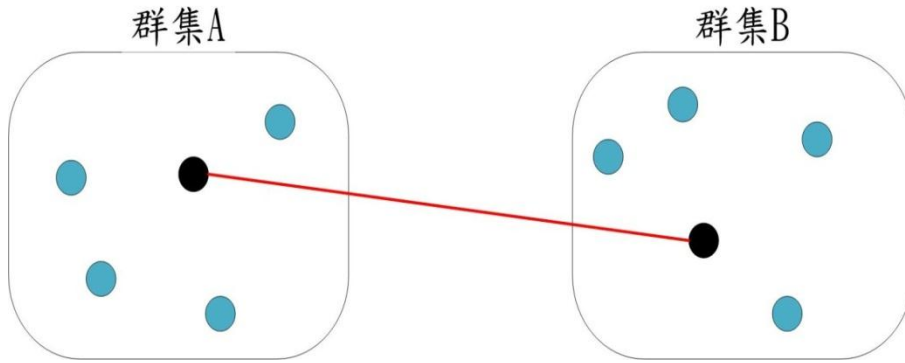


圖 11 平均鏈結法

2.2.3 基於密度分群演算法(Density-base Clustering Algorithms)

以散佈在空間中資料點的密度為主，以密度高的資料點視為同一群，符合分群的概念，同群的資料點相似度高，群與群間相似度低，而常用計算密度的方法：Square wave function、Gaussian function 等...方法，作為門檻值計算某一區塊如果大於門檻值及歸類為一群，代表的演算法有：DBSCAN[30]、OPTICS[34]。

2.2.4 基於網格分群演算法(Grid-base Clustering Algorithms)

將空間劃分成一格一格的網格狀(cell)，計算的方法是以網格為單位分群，可以省下計算群集所耗費的時間成本，依照資料的大小可以調整網格的大小，增加分群的效能，代表的演算法有：STING[31]、CLIQUE[35]。

2.2.5 基於模型分群演算法(Model-base Clustering Algorithms)

該演算法的基礎是建立在假設每一群集都有一個適當的數學模型，使用的數學模型有：統計方法、類神經網路...等，利用這些數學模型找出合適的群集，代表的演算法有：COBWEB[32]。

2.3 Android 系統

Android 系統是由網路界的巨擘 Google 所推廣發展，而在這數位化網路的時代，能連上網路的移動式手持裝置已不稀奇，能提供多樣化的服務才是目前的主流應用，而 Android 就是為了使移動式手持裝置能提供多樣的服務，由 Google 推廣的移動式手持裝置作業系統，因 Google 於 2007 年釋出 Android SDK 讓許多對移動式手持裝置開發有興趣的使用者，可以免費獲得 Android 開發環境設計出不同的應用服務，讓 Android 的市占率快速成長，因為 Android 的開放架構與快速的成長；搭配的硬體需求也越來越多樣化，Google 不得不快速更新 Android 的版本以搭配需求的變化如表 1 所示：

表 1 Android 發行版本

版本	發行日期	改進功能
1.5	2009 年 4 月 30 日	增加支援立體聲藍芽耳機 增加來電顯示照片功能 增加錄影上傳 YouTube 功能 增加圖片(隨手機)橫放轉換角度功能 增加複製/貼上功能 增加虛擬鍵盤 增加選字提示功能 改善 GPS 性能 改善使用者介面 改善相機啟動速度美化視窗邊框
1.6	2009 年 9 月 15 日	增加支援更多螢幕解析度 增加支援手勢 增加支援 CDMA 網路 增加文字轉換語音系統 增加支援 VPN 增加支援 OpenCore2 媒體引擎

		增加快速搜尋框 增加 Google Voice 搜尋 增加應用程式耗電量顯示 增加視聽障礙協助外掛程式 改善拍照介面
2.0/2.1	2009 年 10 月 26 日	增加支援微軟 Exchange 增加支援 HTML5 增加支援閃光燈 增加相機功能(數位變焦) 增加路口轉彎提示導航服務 改善 Google Maps3.1.2 優化硬體運算速度 支援 Google Translate
2.2	2010 年 5 月 20 日	增加藍芽語音撥號 Linux 版本更新至 2.6.32 增加記憶體效能(大於 256MB) 使用 OpenGL ES 2.0 調整 3D 效果 修正相機鏡頭，能在夜間拍照較明亮 修正 Android Market 的偵錯功能
2.3	2010 年 12 月 7 日	改善 UI 提昇使用效率 改善文字輸入模式能更加直覺 修改段落與單字複製功能 改善電源管理模式 增加 SIP 網路電話功能 增加 NFC 近場無線通訊功能

在手持式移動裝置上音樂是主流的服務之一，本篇論文以 Android 為操作背景，開發可以連結雲端音樂服務推薦系統，讓使用者更能享受帶著跑的音樂推薦服務系統。

2.3.1 JSON 資料交換

JSON(Javascript Object Notation)是一種輕量級的資料交換格式語言，傳統的作法是不同的主機與裝置要進行資料的溝通與交換，仰賴的都是 XML(eXtensible markup Language)技術，XML 是一個廣泛應用在各種網路技術與軟體開發的標記語言，但做一般網路資料的數據傳輸，會因為過多的標籤而占去大部分的頻寬，因為在傳輸資料過程中資料的內容並沒那麼的多，可是卻會花費許多標籤去包裝資料的內容，而且在解析端會因為標籤的自定義，造成開發人員為了解析資料而花費更多的時間。JSON 是為了解決網路資料傳輸而發展出的語言，JSON 的包裝方法非常直覺，就是以[Key: Value]作為對應，以“{”作為開始符號，以“}”作為結束符號，而[Key: Value]間以“、”做為區隔如圖 12 所示，因為 Android 與雲端音樂服務推薦系統伺服器間的頻寬有限，而必須有效率的使用，基於 JSON 的特性能更有效率的使用頻寬避免造成浪費，以 JSON 的方式包裝能降低對雲端音樂服務推薦系統伺服器的負擔，而不用在以 XML 的冗長敘述來包裝資料，進而降低負載。JSON 技術目前已經被 Google 列為標準，包含在 Android SDK，提供給開發者自由使用，讓開發者可以自由地運用 JSON 所帶來的輕巧與方便。

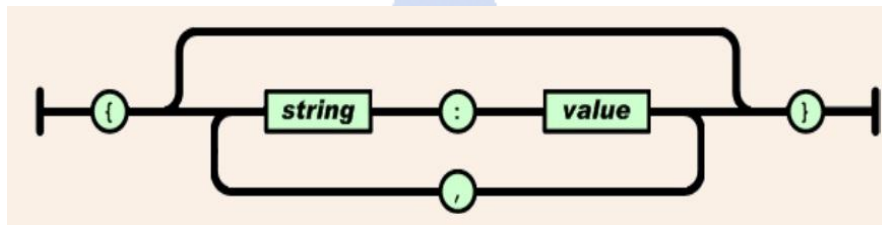


圖 12 JSON 格式

(資料來源：<http://www.json.org/>)

第三章 Android 雲端音樂服務推薦系統建置

本章節介紹雲端音樂推薦系統與 Android 移動式手持裝置的系統建置，3.1 節會介紹整個音樂推薦服務系統的架構與流程；3.2 節介紹雲端音樂推薦系統的音乐特徵以及使用者輪廓的資料定義；3.3 節介紹將使用者分群的 K-means 演算法與改良式 K-means 演算法；3.4 節介紹音樂推薦系統的推薦模組如何將結果呈現給使用者；3.5 節介紹設計 Android 移動式手持裝置的 App 與雲端音樂推薦系統的連結。

3.1 音樂推薦服務系統流程與架構

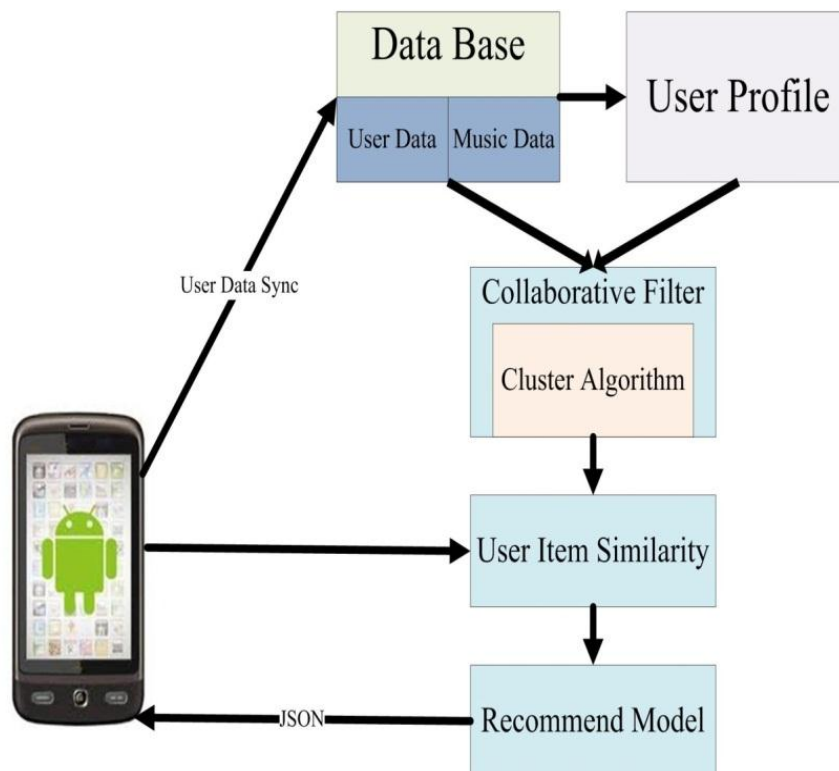


圖 13 音樂推薦服務系統流程

圖 13 是系統的架構圖，系統一開始 Android 移動式手持裝置會與雲端音樂推薦系統進行使用者的連線匹配，接著從資料庫提取使用者的評分記錄與音樂資訊，分析使用者對各類型音樂曲風的喜好，建立使用者輪廓(User Profile)。本系

統將會以協同式過濾的概念為主，將使用者輪廓作為對音樂喜好的依據進行分群演算法的運算，將結果分成 K 群。依照協同過濾的概念，同一群內使用者對於音樂曲風喜好相似度高，系統會將同群使用者的聆聽記錄加入推薦的音樂項目，把可能列為推薦項目的音樂與使用者聆聽過的音樂進行 PCS (Pearson Correlation Similarity)[3][4]運算，運算結果將會產生-1~1 的區間，1 代表相似度高，將 PCS 運算所產生的相似結果，系統將會依照推薦評分公式所產生的預測評分，評分高的音樂優先推薦給使用者。

雲端音樂推薦系統產生對使用者音樂喜好的結果，並將結果以 JSON 的資料格式封裝，並以 Socket 回傳至 Android 移動式手持裝置，Android 移動式手持裝置接收到檔案後，剖析以 JSON 封裝的檔案，並將推薦結果呈現在手機的畫面上供使用者點選。

3.2 建置 Music Style

建置 User Profile(使用者資訊特徵)時需先定義系統應擷取的特徵，本系統是以使用者對於聆聽過的音樂的評分對應至音樂資訊(Music Information)裡的音樂曲風(Music Style)，所以需事先定義音樂曲風的特徵，定義的特徵如表 2 所示；

表 2 音樂曲風特徵

英文曲風	中文曲風
Pop Style	流行曲風
Hip-Hop Style	嘻哈曲風
R&B Style	節奏藍調曲風
Rock Style	搖滾曲風
Metal Style	金屬風格
Gospel Style	福音音樂曲風
Jazz Style	爵士曲風
Dance Style	舞曲風格
Pop-Rock Style	流行搖滾曲風

Rap Style	饒舌曲風
Hard-Rock Style	重搖滾曲風
Punk Style	龐克曲風
Folk Style	民族風搖曲風
New-Age Style	樂器旋律風格
Electroponic Style	電子音樂曲風
Reggae Style	雷鬼曲風
Other-Style Style	其他音樂曲風

表 2 是顯示本系統所定義的音樂曲風共 17 種，系統依照制定的曲風將使用者對聆聽過的音樂評分進行計算，依定義的曲風計算出使用者對各曲風的分數，建立一組代表使用者喜好的 User Profile，而該 User Profile 是以音樂曲風作為使用者對於音樂的喜好程度，所以本系統將會產生一組 17 維度，代表使用者對於音樂喜好程度的 User Profile。

本系統採取音樂的曲風做為對使用者聆聽音樂喜好的依據，是因為作者認為聆聽喜好的音樂跟語言與男女無關，假設使用者 A 今天喜歡聆聽搖滾音樂，可是他只會聽亞洲市場的搖滾音樂曲風嗎？並不盡然，使用者 A 一定有很大的興趣想嘗試聆聽歐美市場，或其他創作歌手的搖滾曲風，所以本系統採取音樂曲風做為使用者對於喜好類型的音樂定義。

3.3 建置 User Profile

雲端音樂推薦系統在對使用者的喜好進行推薦之前，系統必須要掌握使用者一定程度的喜好。以本篇論文的系統來說，系統資料庫裡已經存放使用者的相關資料以及音樂的相關資料，其中使用者的相關資料除了包含使用者的基本資料外，還存放著使用者聆聽過的音樂記錄，以及使用者對聆聽過音樂的評價，該音樂評價是使用者以顯性評分(Rating)的方式記錄著，評分的區間是 0~100 分，評分的高低代表使用者對於聆聽過的音樂喜好，而雲端音樂推薦系統就是根據使用者對於聆聽過的音樂評分作為使用者對於音樂喜好的推薦依據。

系統會將使用者的評分統計，並分析使用者聆聽過的音樂，以及對該音樂的評分，對應出該音樂的音樂曲風(Music Style)，進而找出使用者對於聆聽音樂曲風的喜好，如圖 14 所示；

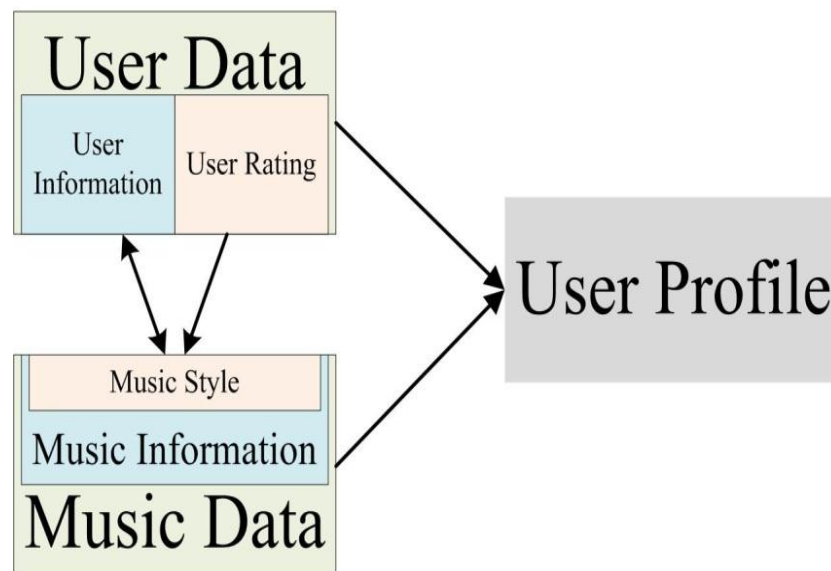


圖 14 User Profile 建置流程

圖 14 是 User Profile 建置流程，圖中顯示 User Rating 與 Music Style 之間的關係與資料庫的結構，收集這兩者的特徵建立出一組使用者對於音樂曲風的喜好，該音樂曲風的喜好代表著使用者所喜好的音樂類型，建立 User Profile 的目的是以推薦的目標：音樂，與推薦的服務對象：使用者，兩者間建立起一組可表示關聯的特徵，再依照協同式過濾的概念，找出具有相似喜好的使用者，再參考其他具有相似喜好的使用者聆聽紀錄推薦給使用者。

3.3.1 User Profile 的計算

定義出對使用者喜好的音樂曲風，接下來就要分析使用者聆聽音樂的紀錄與評分，產生出一組 17 維的音樂曲風向量，假設使用者為 U ，則對使用者的音樂曲風向量為：

$$U^i = \{S_x, x = 1, 2, \dots, 17\}, i \text{ 表示為每一位使用者的編號, } x \text{ 表示每一維}$$

度的音樂曲風，如公式 3.1 所示。

$$U_x^i = S_x = \frac{\sum_{m=1}^n R_m^x}{n^x}, \forall x = 1 \sim 17 \quad (3.1)$$

上述的公式符號分別為：

U：表示使用者。

x：表示目前計算的音樂曲風維度， $x = 1 \sim 17$ 。

S_x ：表示使用者在該維度音樂曲風的平均評分。

R：表示使用者的評分。

i：表示目前是第 i 位使用者。

m：表示使用者在該維度的音樂評分筆數，共有 $1 \sim n$ 筆。

系統經由上述的公式產生出使用者對於音樂曲風的向量表示表，每組曲風的向量彼此不受干擾，在進行正規化計算的時候，本系統採取的是每組維度皆為獨立表示，在這裡指的是每組維度都有各自的評分次數，在進行正規化的平均評分為計算該維度的總評分和除以評分次數，在正規化的部份，不同的系統有不同的正規化方式，有的系統是採取最大化評分方式，以評分次數最高的曲風做為正規化的標準，好處是經過正規化後能將喜好的特徵凸顯出來，但是相對的，如果其他維度的特徵評分次數低，結果將被稀疏化，這樣對於其他維度的成長將會受到影響，如表 3、表 4 所示；表 3 為本篇論文所採取的正規化方法，各個維度皆為獨立，皆以各自的總評分除以評分次數，得到各維度的正規化結果，但從表 4 採取最高評分次數做為正規化標準，可以看出紅色的區塊表示該維度已被稀釋，可是從該維度的總評分與評分次數來看，該使用者對該維度的興趣並沒有那麼低，所以本系統採取各維度的正規化做法，以維度 1 為例：使用者對該維度的評分次數不高，表示使用者對該維度曲風的音樂有著推薦的價值，擴展使用者對該維度的發展性。

表 3 各維度正規化

	維度 1	維度 2	維度 3
總評分	2	20	10
評分次數	3	22	20
正規化	0.66	0.9	0.5

表 4 最高次數正規化

	維度 1	維度 2	維度 3
總評分	2	20	10
評分次數	3	22	20
正規化	0.09	0.9	0.5

3.4 User-base 分群演算法

上節提到雲端音樂推薦系統在進行推薦前，系統須先了解使用者對各個音樂曲風的喜好，系統將會建立一組使用者對各音樂曲風喜好的向量模型，接著系統進入將使用者分群階段；系統將依據先前建立的使用者音樂曲風向量，計算出全部使用者中彼此相似的使用者，根據上一章節介紹的協同過濾的概念，同群使用者彼此對音樂的喜好接近，所以其他相似使用者聆聽過的音樂，都適合用來推薦給使用者。如何找出相似的使用者？本系統以 **K-means** 分割式分群演算法為基礎，改良 **K-means** 演算法的缺點，以本系統提出的改良式 **K-means** 演算法為基礎找出相似的使用者。

圖 15 為本篇論文提出的改良式 **K-means** 分群演算法流程，在開始進行改良式 **K-means** 分群演算法前，系統會開始初始化所有資料，初始化完成後便進入初始化群中心的階段。改良式 **K-means** 演算法第一步就是找出能代表該資料空間的資料點，以這些資料點作為初始化群中心的基準，找出合適的初始化群中心進行分群。進入分群的步驟，系統會開始計算資料空間中所有的資料點與各群中

心間的相似度關係，將資料點分配給相似度高的群中心，最後產生新的群中心，如果新的群中心與原來的群中心是不同的，則以新的群中心為搜尋相似資料的標準再重新分群，這過程稱為“迭代”，而迭代的結束則是直到群中心不再發生變化為止。分群結束後開始評估群集內資料點與群中心間的相似度，如果其中一資料點與群中心的相似度值低於門檻值，代表該資料點與群中心的相似度低，系統將會重新分群，並將 K 值增加提升群集的密度，之後的小節會介紹傳統 K -means 分群演算法，並說明其缺點，之後介紹本篇論文提出的改良式 K -means 分群演算法的改良分法。



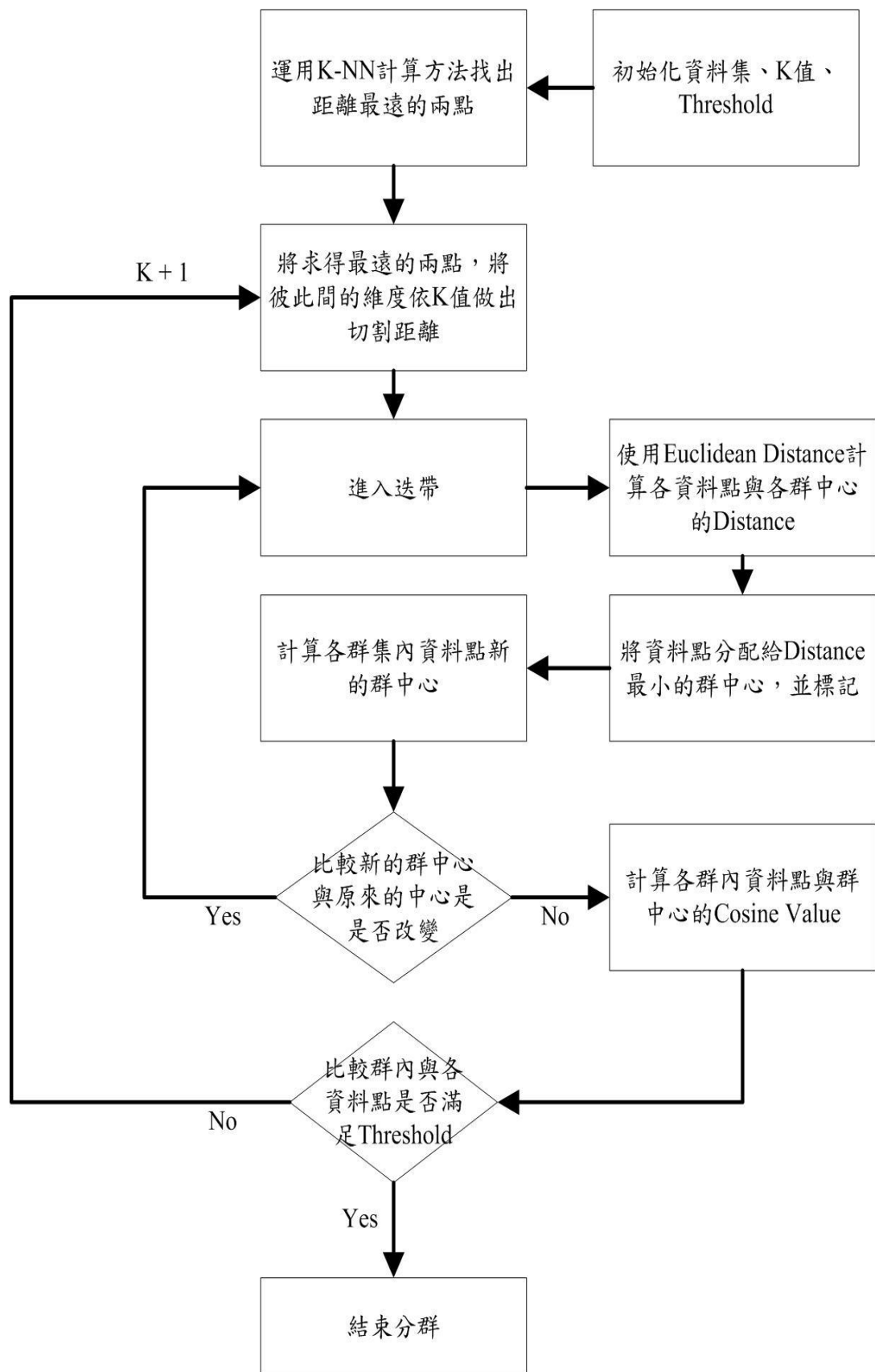


圖 15 改良式 K-means 演算法

3.5 K-means 分群演算法

K-means 演算法是由 J. MacQueen[2]所提出，其目的是找出散佈在空間中的資料點，依使用者指定的群數 K ，將彼此相近的資料劃分為 K 群，K-means 的分群方式是以群重心為主，其演算流程如圖 16 所示：

- 一、使用者指定欲分的群數 K 。
- 二、從資料點中亂數選取 K 點作為初始群中心。
- 三、計算各資料點與各群中心的距離，距離最小者將分配至該群集。
- 四、計算新的群中心。
- 五、與原來群中心做比較如果不同則更新群中心重複步驟二。

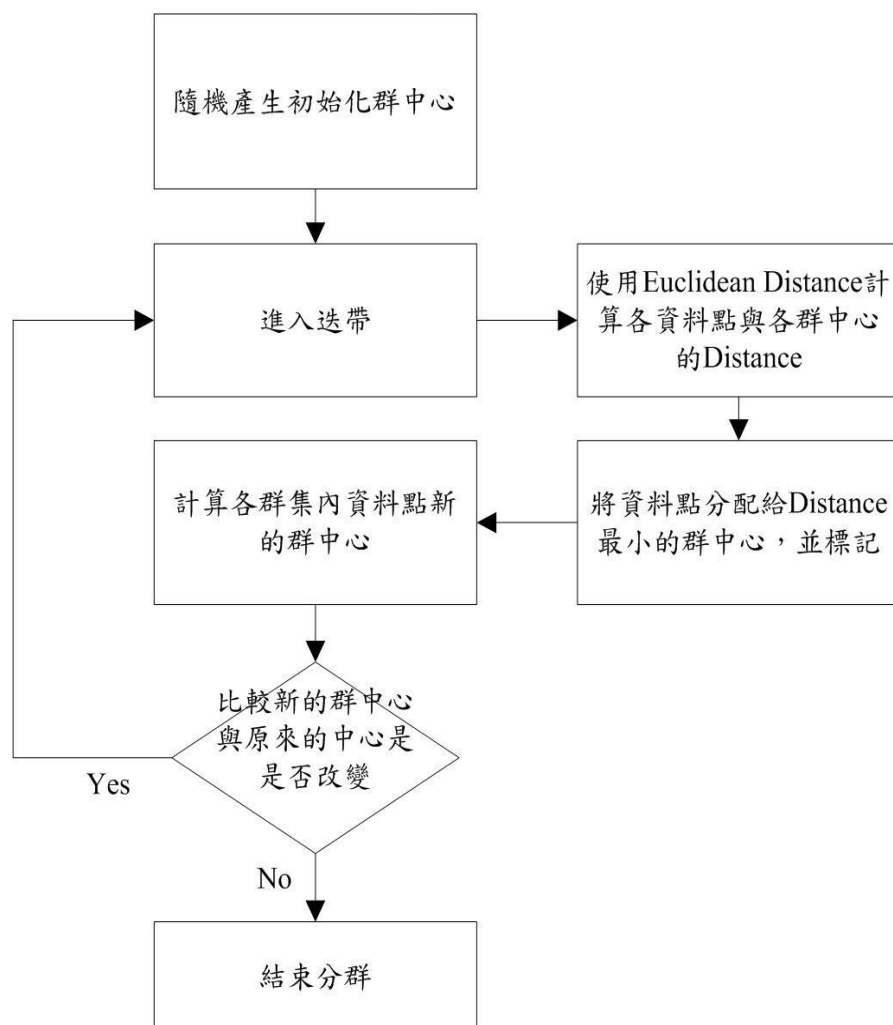


圖 16 K-means 演算流程

經過多次的迭帶，群中心將不斷的移動，最後的結果就是將資料點歸屬於與群中心相似的群，K-means 是一套發展許久又受多人使用的演算法，可是從上述的演算流程來看，有幾種會影響到 K-means 的分群結果的因素：

- 一、隨機產生的初始化群中心。
- 二、使用者須指定分群的群數 K。
- 三、衡量資料點與群中心的關係。

上述幾點有許多的學者做過不同的改良方式，B. Chen[16][29]提出結合階層式分群演算法的概念用來改良初始化群中心的方法，其他學者也針對初始化群中心做出不同的改善[14]，D.M. Kim[17]針對 K-means 分群的分群數做出改良，在分群過程中加入了群半徑的概念，當群的半徑過大在分裂，直到全部的群半徑小於事先設定好的半徑，本系統提出改良式 K-means 分群演算法，該演算法的目的就是針對上述的缺點進行改善，增加同群內使用者的相似性。

3.6 改良式 K-means 分群演算法

本篇論文提出改良式 K-means 分群演算法，用來解決亂數決定初始群中心而造成分群結果不佳，並改善使用者須自行輸入 K 的數值，由分群過程來決定該資料群適合的群數。

3.6.1 改良初始化群中心

K-means 初始化群中心問題，一直都是許多學者想改善的目標，所以如何選出好的初始化群中心對 K-means 之後的分群結果有很大的影響，本篇論文提出的改良初始化群中心問題是基於 K-NN 演算法(K-th Nearest Neighbor Algorithm)[5][18]發展而來，K-NN 的中文名稱又稱為“第 K 個鄰近搜尋法”，例如：資料點 A 散佈在一群資料點中，而我們要找出離資料點 A 最近的點就稱作“第 1-th”，以此類推離資料點第二近的稱為“第 2-th”。K-NN 演算法是一種非監督式的學習演算法，不需要經由資料訓練，只需將資料輸入就會產生結果，

而在 K-NN 搜尋全域最鄰近點過程中一定會找出最遠的點如圖 17 所示。

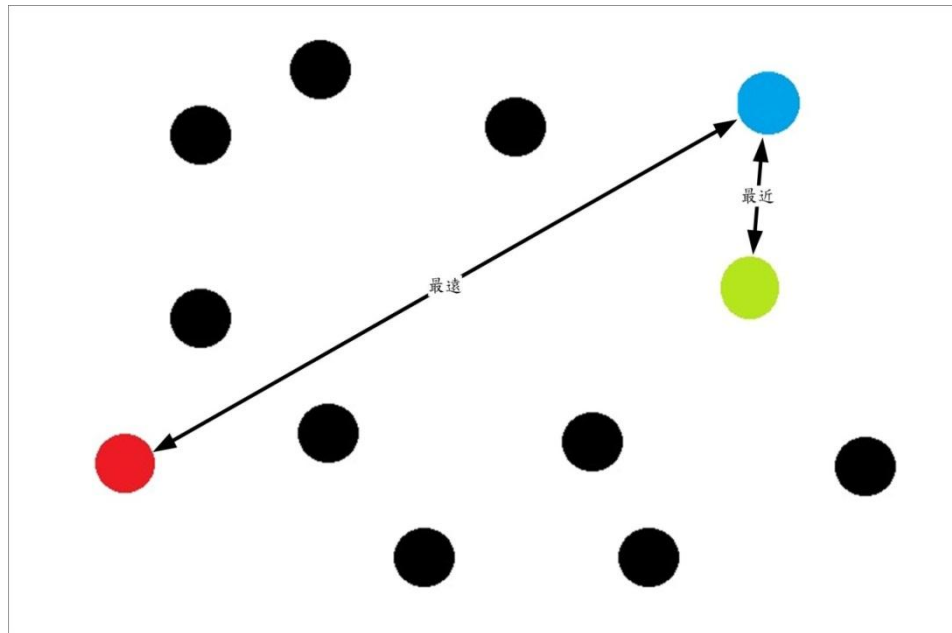


圖 17 K-NN 搜尋最遠、最近資料點

本篇論文是以搜尋出散佈在資料空間中最遠的兩點為目標，為什麼要找出最遠的兩點？是因為數值資料中最遠的兩點代表的是在全部的資料點中該兩點彼此差異最大，即該兩點的數值囊括資料空間中絕大部分的資料點與數值範圍，即表示該兩點在該資料空間中是較極端的數值，如公式 3.2：

$$\text{The worst data} \leq S \leq \text{The best data}, S \in \text{All data} \quad (3.2)$$

本系統找出最遠兩點的方法是運用 Euclidean distance 計算出各點間彼此的距離，如公式 3.3：

$$d(p, q) = \sqrt{\sum_{i=1}^n (p_i - q_i)^2}, i = 1 \sim n \quad (3.3)$$

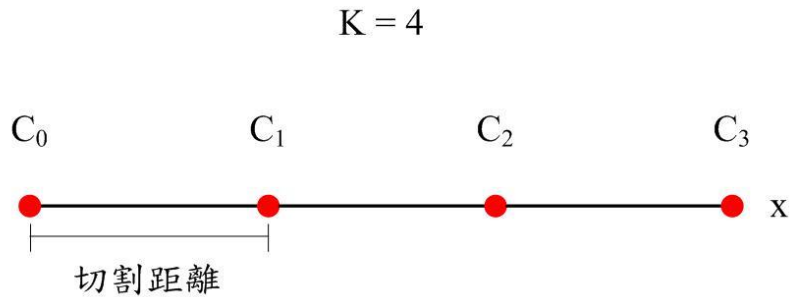
上述的公式符號分別為：

p、q：是全部資料點中的任兩點。

n：為資料點的維度。

d：兩點資料點的 Euclidean distance。

找出最遠的兩點接著將資料的各別維度進行切割，根據欲分群數計算出各群中心間各維度的距離，再以分群數決定各維度距離的切割數，該切割數即為新的群中心數量，而各群中心維度的距離必須平均，所以先計算兩個維度間的切割距離，而切割距離即鄰近群中心維度間的距離如圖 18 所示，切割的方法如公式 3.4：



$$n_{cut} = d_{cut}^i = \frac{p_i - q_i}{(K-1)}, i = 1 \sim n, K > 2 \quad (3.4)$$

上述的公式符號分別為：

K：分群數。

n：為資料點的維度。

d：為切割距離。

找出切割距離開始計算初始群中心的位置，以本系統為例：一開始分群數 $K=2$ ，系統就不需計算切割距離以最遠兩點作為初始群中心，進行迭帶。如果 $K>2$ ，則需以切割距離作為決定新初始群中心的計算，前一章節說明本系統的每組維度皆為獨立，而計算出的切割距離一樣皆為獨立，每組維度皆有屬於自己的切割距離，計算新初始群中心公式，如公式 3.5：

$$C_i = C_{i-1} + d_{cut}^i, i = 2 \sim (n - 1) \quad (3.5)$$

上述的公式各符號的說明如下：

p、q：表示最遠的兩點。

i：表示資料的維度。

d_{cut}^i ：表示該維度的切割距離。

K：表示群數。

C：表示群中心。

經過計算後產生的初始群中心，以表 5 為例：

表 5 K-NN 初始群中心			
K = 4	維度 1	維度 2	維度 3
P	60	10	30
q	3	70	45
切割距離	19	20	5
計算並產生初始群中心			
K = 1	60	10	30
K = 2	41	30	35
K = 3	22	50	40
K = 4	3	70	45

表 5 表示 p、q 兩個維度大小各為 3 的資料點，系統欲分 4 群，經由公式計算產生各維度間彼此的切割距離，再依各維度的切割距離產生新的初始群中心。

運用 K-NN 演算法找出散佈在資料空間中最遠的兩點，再依分群數決定切割距離的大小，最後產生從兩點極端的數值資料囊括大部分資料的初始群中心，初

始化群中心系統流程如圖 19 所示。

本篇論文提出改良式 K-means 分群演算法，用以改善傳統 K-means 分群，因為隨機選擇初始群中心而造成的分群結果不佳，利用數值極端的兩點與分群數決定切割距離調整出可以包含大部份數值資料的初始群中心。

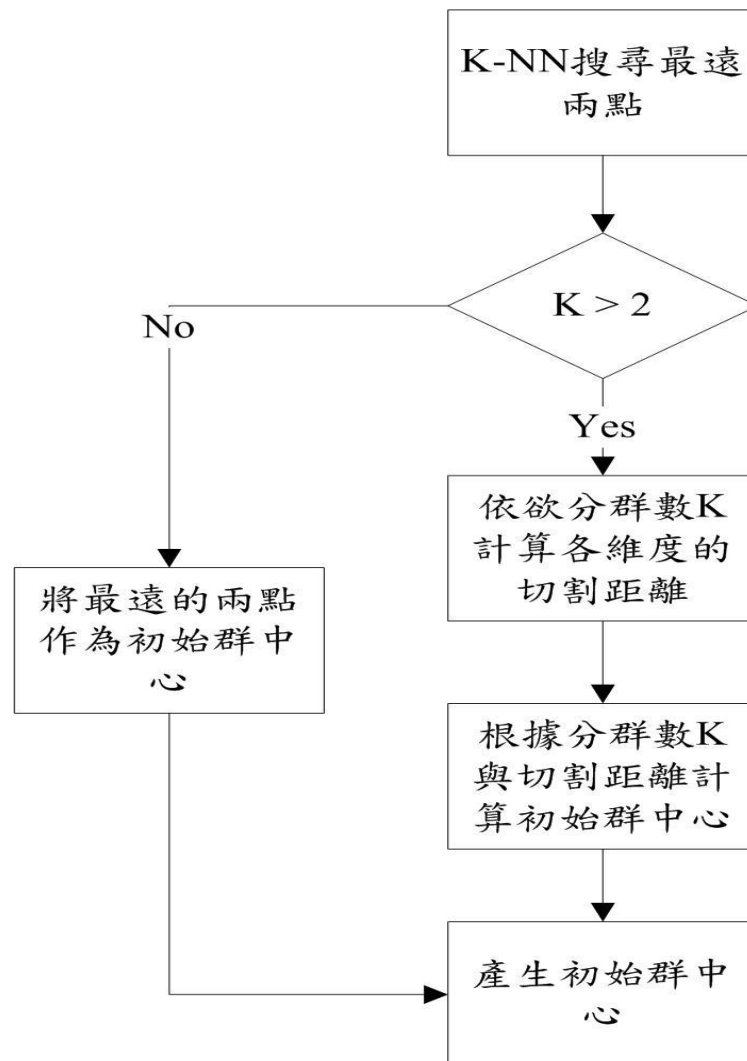


圖 19 初始化群中心流程

3.6.2 動態 K 值演算法

在上節本篇論文提出改良 K-means 初始群中心問題，運用 K-NN 演算法分析出合適的初始群中心，在本節將針對使用者須指定 K 值(欲分群數)與衡量群中心和資料點間的關係提出改良的演算法。

傳統的 K-means 分群法使用者須指定分群數 K 值進行分群，接著再以 Euclidean distance 計算群中心與資料點間的關係，但是使用者往往並不清楚欲分析的資料存在著甚麼關聯，而造成分群的結果並不理想。衡量群中心與資料點的 Euclidean distance 並不適合所有類型的資料型態，僅能分析數值化的資料，非數值化的資料例如：顏色、語意、性格、等...，無法以數值表示高低或相關程度的資料，所以非數值化資料不在本篇論文的討論範圍內，Euclidean distance 是常用來衡量數值資料相似度的計算公式，在使用上簡單好處理，並且在低維度空間有著滿不錯的準確度，但是資料擴展至高維度空間，在衡量彼此相似度的過程中，很容易受到其他維度的影響，而造成衡量彼此間相似度的準確度降低，增加分析資料的錯誤率，如表 6 所示：

表 6 三維範例

	維度 1	維度 2	維度 3
資料 A	8	4	6
資料 B	8	3	5
資料 C	4	2	3

由表 6 可以看出資料 A 與資料 B 彼此是相似的；資料 A 與資料 C 彼此間是不相似的，經由 Euclidean distance 計算得到資料 A 與資料 B 的距離為：1.4；資料 A 與資料 C 的距離為：5.3，根據觀察與公式的驗證皆符合先前的推論；資料 A 與資料 B 是相似的，資料 A 與資料 C 是不相似的，但是將維度擴展為更高的維度結果是否與前一例子一樣，如表 7 所示：

表 7 五維範例

	維度 1	維度 2	維度 3	維度 4	維度 5
資料 A	8	4	6	9	3
資料 B	8	3	5	5	9
資料 C	4	2	3	7	5

由表 7 將三筆資料的維度擴展至維度 5；其中維度 1～維度 3 的值與先前一樣不變，經由觀察得到資料 A 與資料 B 是相似的；資料 A 與資料 C 是不相似的，經由 Euclidean distance 計算得到資料 A 與資料 B 的距離為：7.3；資料 A 與資料 C 的距離為：6.0，Euclidean distance 產生的結果顯示資料 A 與資料 C 是相似的，資料 A 與資料 B 是不相似的，觀察與公式產生的結果發生矛盾，如果資料的維度繼續向上擴展，部份維度的值將會影響整體資料相似度判斷的結果將會越明顯。

上一段說明了 Euclidean distance 在高維度的資料空間進行相似度的計算很容易受到部份維度的值影響，這是因為 Euclidean distance 無法有效的鑑別各維度間的變化關係，所以本篇論文引用另一組公式[19]；餘弦相似度(Cosine similarity)：如公式 3.6 所示

$$\cos \theta = \text{Sim}(a, b) = \frac{\vec{a} \cdot \vec{b}}{|\vec{a}| |\vec{b}|} = \frac{\sum_{i=1}^n a_i * b_i}{\sqrt{\sum_{i=1}^n a_i^2} \sqrt{\sum_{i=1}^n b_i^2}} \quad (3.6)$$

餘弦相似度是在空間向量中是很常用來衡量彼此相似度的衡量公式，對於各維度間的變化餘弦相似度能很敏感的發覺，衡量兩者間相似度的值為-1～1 區間，數值越大代表兩者間的相似度越高，如圖 20 所示：

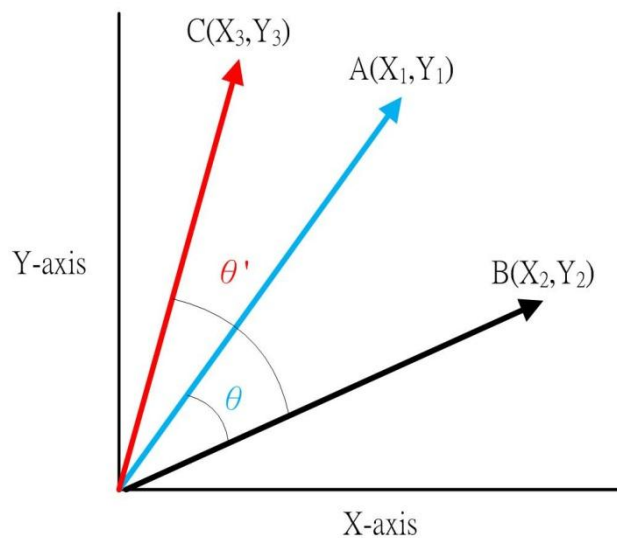


圖 20 餘弦相似度

A 與 B 間兩者的夾角 θ 越小，餘弦相似度計算的結果越高，表示兩者越相近，但是餘弦相似度的計算上能存在著缺點，例如表 6：我們知道資料 A 與資料 B 是相似的；資料 A 與資料 C 是不相似的，但是由餘弦相似度判斷；告訴我們資料點 A 與資料點 B 的值為 0.9，資料點 A 與資料點 C 的值為 1，結果告訴我們資料點 A 與資料點 C 是相似的，這與先前的結果發生矛盾，但是在 Euclidean distance 卻沒有這種問題，所以得到一個結論如表 8 所示：

表 8 計算相似度比較

	優點	缺點
Euclidean distance	對於整體維度的判斷佳，在維度越低的空間衡量兩筆資料的相似度越準確。	容易受到部份維度的影響，造成相似度判斷降低。
Cosine similarity	對於各別維度判斷相似度的準確度佳，且不受維度空間大小的影響。	對於整體維度的判斷，容易受到維度空間線性相依的影響，造成相似度判斷降低。

綜合上述的結果本篇論文結合 Euclidean distance 與 Cosine similarity 各自的優點，採用循序法計算資料與群中心間的相似度，用以調整分群數 K 值，如圖 21 所示：

一開始先初始 K 值(本雲端音樂推薦系統是以最小群數 $K=2$ ，進行分群)決定分群數與設定門檻值(Threshold) θ ，使用 Euclidean distance 計算資料點與群中心的距離，分群結束後開始計算群集內分配到的資料點與群中心的相似度，運用餘弦相似度計算群中心與資料點的相似度值，判斷相似度值是否符合門檻值如圖 22 所示，如果不符合表示群內有資料點與群中心的相似度低，則改變 K 值提高分群的密度“重新分群”，直到相似度值符合門檻值分群即結束。

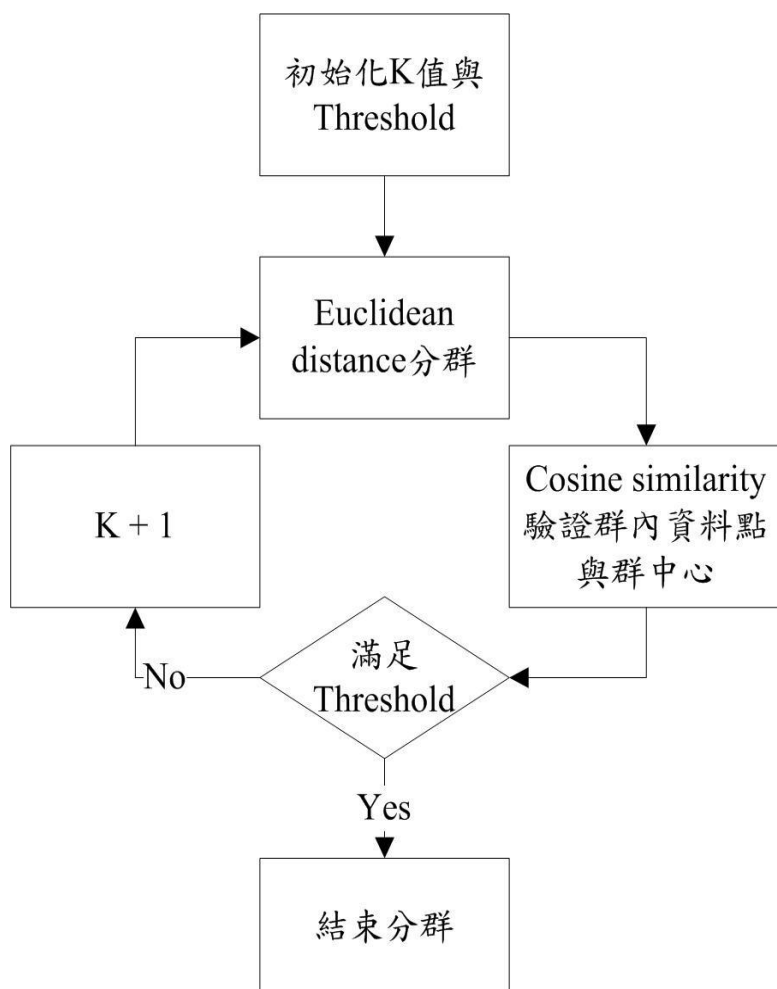


圖 21 餘弦門檻分群

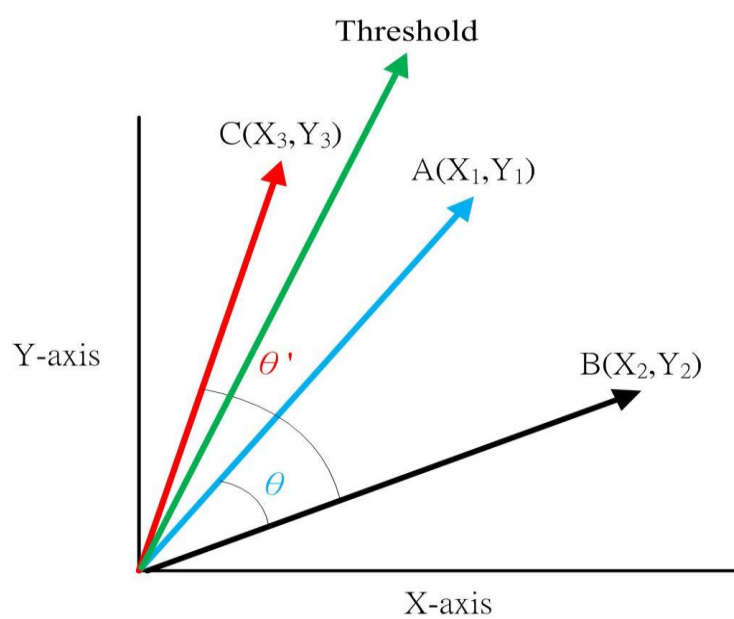


圖 22 餘弦門檻值

3.7 協同式過濾音樂推薦模組

上一節使用改良式 K-means 分群演算法將彼此相似的使用者歸類為同群，本篇論文採用協同式過濾的概念，所以同群的使用者彼此都是具有類似喜好的使用者，系統會將同群內使用者聆聽過的音樂成為推薦給使用者的音樂推薦項目。

既然同群使用者具有類似的音樂喜好，那系統推薦給使用者的音樂項目就從這些同群使用者挖掘出來，本系統對每一筆欲推薦的音樂項目會使用預測公式 [3][4][20]產生預測評分，預測評分高的音樂項目就推薦給使用者，預測評分公式如公式 3.7

$$P_{a,j} = \overline{v_a} + k \sum_{i=1}^n w(a,i) (v_{i,j} - \overline{v_i}) \quad (3.7)$$

$P_{a,j}$ ：使用者 a 對於音樂項目 j 的預測評分。

$\overline{v_a}$ ：使用者 a 的平均評分。

$v_{i,j}$ ：使用者 i 對於音樂項目 j 的評分。

$\overline{v_i}$ ：使用者 i 的平均評分。

k ：正規化係數，設定值為 1。

$w(a,i)$ ：推薦者與被推薦者間的相關係數。

上述公式是本系統推薦預測模組，其中 $w(a,i)$ 是推薦者與被推薦者間的對於音樂項目的喜好程度，考慮到推薦的音樂項目與推薦者、被推薦者間的關係，使用皮爾森相關係數(Pearson correlation similarity)衡量推薦者、被推薦者與推薦音樂項目間的關係，如公式 3.8：

$$w(a,i) = \frac{\sum_j (v_{a,j} - \overline{v_a})(v_{i,j} - \overline{v_i})}{\sqrt{\sum_j (v_{a,j} - \overline{v_a})^2 \sum_j (v_{i,j} - \overline{v_i})^2}} \quad (3.8)$$

經由皮爾森相關係數計算推薦者、被推薦者與推薦音樂項目三者間的關係係數，其中 $v_{a,j}$ 與 $v_{i,j}$ 是推薦者與被推薦者的音樂評分，將皮爾森相關係數的值結合

預測模組的計算產生出預測評分，而本系統的推薦方式採取 Top-N 的做法將預測評分高的音樂推薦項目推薦給使用者。

3.8 Android 移動式手持裝置的開發與連結

本篇論文以 Android 移動式手持裝置連結雲端音樂推薦系統進行推薦的服務，開發 Android App 設計使用者 GUI 提供使用者操作，底層以 Socket 連結雲端音樂推薦系統並送出使用者資訊，收到使用者傳來的訊息，雲端音樂推薦系統便會開始進行推薦，並將推薦的結果以 JSON 資料交換格式包裝，傳回 Android 移動式手持裝置如圖 23 所示：

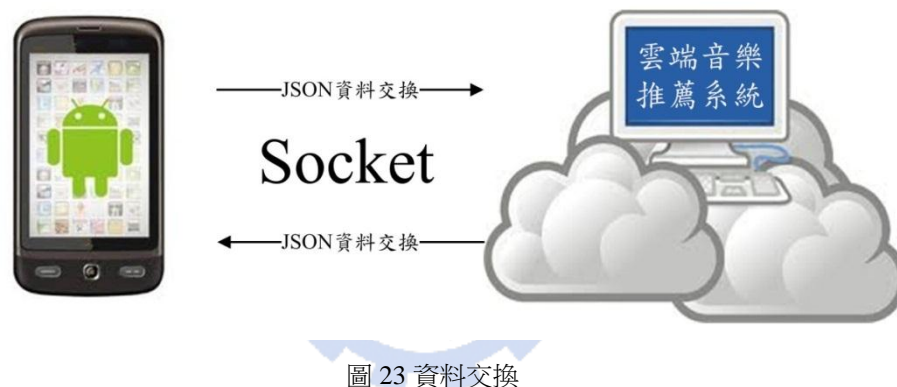


圖 23 資料交換

Android 移動式手持裝置收到來自雲端音樂推薦系統的推薦結果開始解析 JSON 封裝如圖 24 錯誤! 找不到參照來源。所示，解析的語法會將雲端音樂推薦系統伺服器傳回的資訊進行解讀，並由 Android 將雲端音樂推薦系統的推薦結果呈顯在使用者畫面上。

```
JSONObject json = new JSONObject(JsonString);
```

圖 24 Json 解析

第四章 實驗結果與評估

在本章節將說明建置雲端音樂推薦系統的架構與環境，與實驗評估在 4.1 介紹系統環境架構與資料庫；在 4.2 節介紹分群演算法的評估方法，並評估改良式 K-means 分群演算法與其他分群演算法的分群結果；在 4.3 節介紹評估雲端音樂推薦系統的評估方法，評估運用不同分群演算法的推薦結果；在 4.4 節展示系統結果。

4.1 環境建置

本篇論文所使用的系統環境如下：

- 開發語言：JAVA。
- 開發環境：Microsoft windows-7
- 資料庫：MySQL 5.5
- 資料來源：本實驗使用 Yahoo 與 KDD CUP 聯合舉辦的音樂推薦系統比賽所提供的資料集[7]。

原始的資料集包含了 249012 使用者與 62551438 筆評分紀錄，因為原始數據資料太過龐大所以本篇論文只取一部份資料作為系統實作用，資料集內容如表 9 所示：

表 9 資料庫資料集

User	5103
User rating	1349514
Music item	224041

本篇論文將評估 3 種分群結果，分別為：傳統 K-means 分群、改良式 K-means 分群、以 CURE 作為初始化群中心的 CURE-K-means 分群，而為了方便稱呼將改良式 K-means 分群、CURE-K-means 分群分別稱為：M-K-means 與 C-K-means，因為改良式 K-means 分群並沒有隨機的概念，所以分群結果取一次即可，而 K-means 分群與 CURE-K-means 分群有隨機的變數，所以評估結果將各執行 10 次取平均作為評估的數值。

4.2 分群評估

上一章提出改良式 K-means 分群演算法改善初始化群中心問題與使用者指定 K 值，目的是為了提高同群使用者的相似度，而本篇論文將採用凝聚率 (Agglomerate rate) 與鑑別率 (Discrimination rate) [22][23]，做為評估分群的指標。

4.4.1 凝聚率 (Agglomerate rate)

好的分群結果是能夠找出相似資料歸類為同一群，而歸類為同一群的資料相似度能越高越好，以群的概念來看是指資料能夠越凝聚則分群的效果越佳，而凝聚率較高的群其資料間的距離是相當短的，所以評估凝聚率的做法是計算群內所有資料點與群中心間的距離，在標準距離內的資料點越多表示凝聚率越高，如何計算標準距離？在這裡我們使用常態分佈的概念，根據常態分佈的定義有 99.74% 的資料會分佈在 $\mu + 3\sigma$ 範圍內，所有資料點到群中心距離平均 μ 與 $\mu + 3\sigma$ 內所有資料點如圖 25，做為評估凝聚率的標準距離，在 $\mu + 3\sigma$ 範圍的資料越多資料點到群中心的距離都在標準距離 μ 內則凝聚率越高，計算 μ (平均值) 與 σ (標準差) 的公式如公式 4.1、公式 4.2 所示：

$$\mu = \frac{\sum_{i=1}^n |c - c_i^d|}{n}, n \in C \quad (4.1)$$

$$\sigma = \sqrt{\frac{\sum_{i=1}^n (|c - c_i^d| - \mu)^2}{n}}, n \in C \quad (4.2)$$

$$\text{Agglomerate rate} = \frac{\text{在 } \mu \text{ 範圍內的資料點}}{\text{落在 } \mu + 3\sigma \text{ 範圍內的資料點}}, 0 \leq \text{Agglomerate rate} \leq 1$$

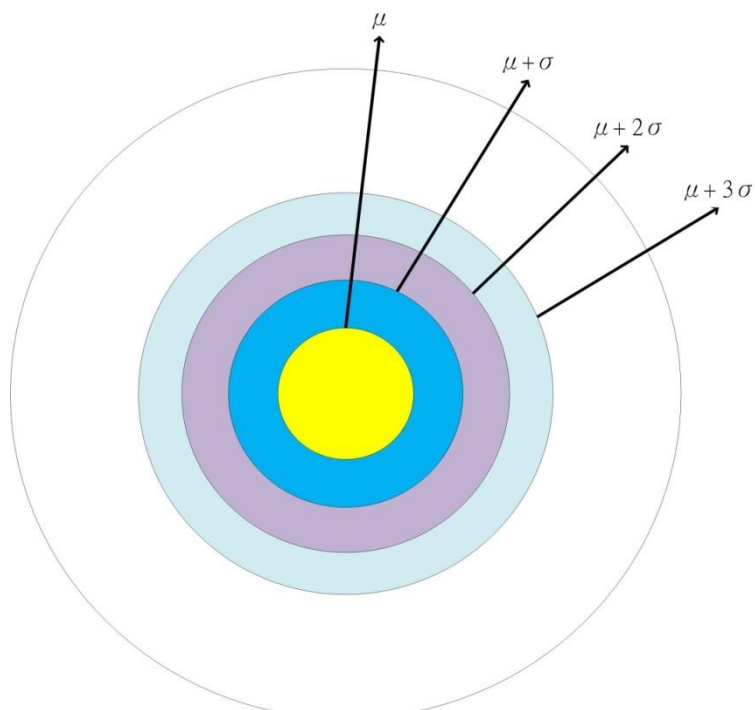


圖 25 凝聚率示意圖

本篇論文將使用鑑別率評估 3 種分群結果，評估的方式以改良式 K-means 為基準設定門檻值，門檻值為 $0.5(\cos 60^\circ)$ 、 $0.7(\cos 45^\circ)$ 、 $0.86(\cos 30^\circ)$ ，經過分群計算分出的結果為：115 群、121 群、181 群，而其他的分群將會以改良式 K-means 所分出來的群數為基準進行分群，如表 10、圖 26 所示：

表 10 凝聚率

	M-K-means	C-K-means	K-means
$0.5(\cos 60^\circ)$	0.57160	0.57023	0.55494
$0.7(\cos 45^\circ)$	0.57509	0.55775	0.55691
$0.86(\cos 30^\circ)$	0.58072	0.56683	0.55746

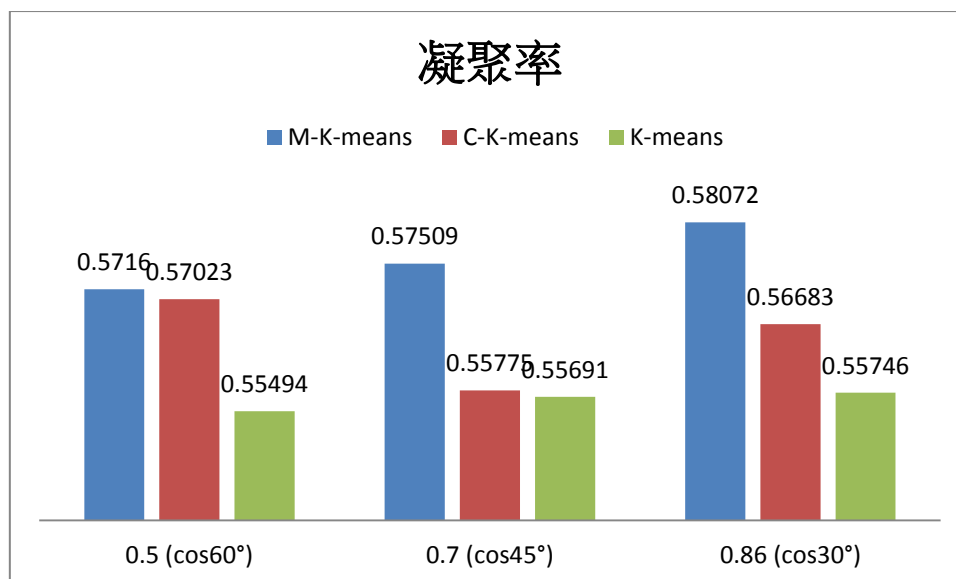


圖 26 凝聚率

由上述的結果可以觀察出改良式 K-means 分群的凝聚率，比其他兩種分群方法好，代表對本篇論文使用的資料集而言，本篇論文提出改良式 K-means 分群演算法，可以有效地搜尋出相似的資料點，提高群集內的相似度，對使用者來說能搜尋出更多相似的資料更佳的提升分群的效能。

4.4.2 鑑別率(Discrimination rate)

好的分群結果是能夠找出相似資料歸類為同一群，群與群之間是不相似的，基於以上的特性，我們可以運用相似的資料為同一群，而群與群之間是不相似的概念鑑別出分群的好壞，而鑑別出分群的好壞就在於如何找出群與群間不相似的程度，不相似的程度越高表示鑑別率越佳，作法是採取兩群間群中心的距離，與兩群間距離最短的點作為鑑別兩群的方法如圖 27 所示，以兩群群中心的距離作為兩群距離的標準，而兩群最短的距離作為鑑別兩群不相似程度的標準，兩群的最短距離越大，代表兩群的不相似程度越高鑑別率越佳，反之兩群的最短距離越小，代表兩群的相似程度越高鑑別率越低，鑑別率的公式如下：

$$\text{Discrimination rate} = \frac{\text{兩群間最短距離}}{\text{群中心距離}}, 0 \leq \text{Discrimination rate} \leq 1$$

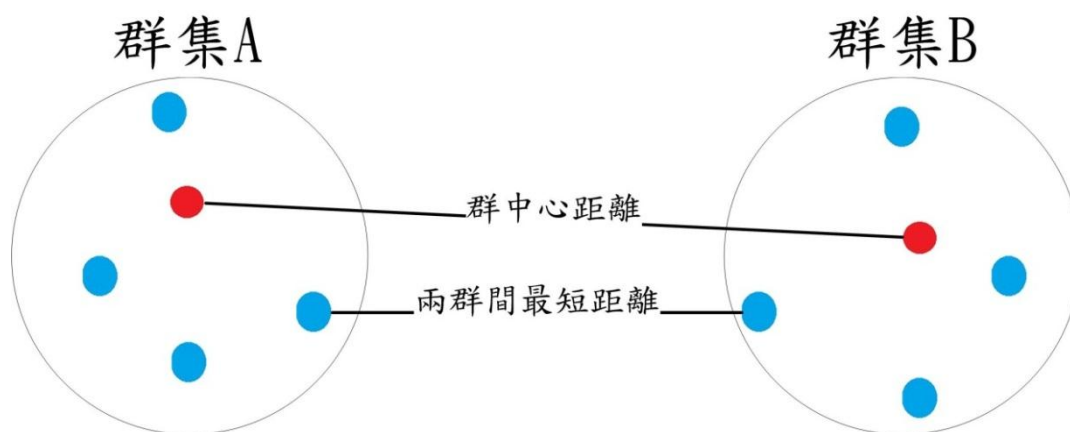


圖 27 鑑別率示意圖

本篇論文將使用鑑別率評估 3 種分群結果，評估的方式以改良式 K-means 為基準設定門檻值，門檻值為 $0.5(\cos 60^\circ)$ 、 $0.7(\cos 45^\circ)$ 、 $0.86(\cos 30^\circ)$ ，經過分群計算分出的結果為：115 群、121 群、181 群，而其他的分群將會以改良式 K-means 所分出來的群數為基準進行分群，如表 11、圖 28 所示：

表 11 鑑別率

	M-K-means	C-K-means	K-means
$0.5(\cos 60^\circ)$	0.75889	0.74849	0.72411
$0.7(\cos 45^\circ)$	0.76615	0.74218	0.73079
$0.86(\cos 30^\circ)$	0.80935	0.77862	0.76956

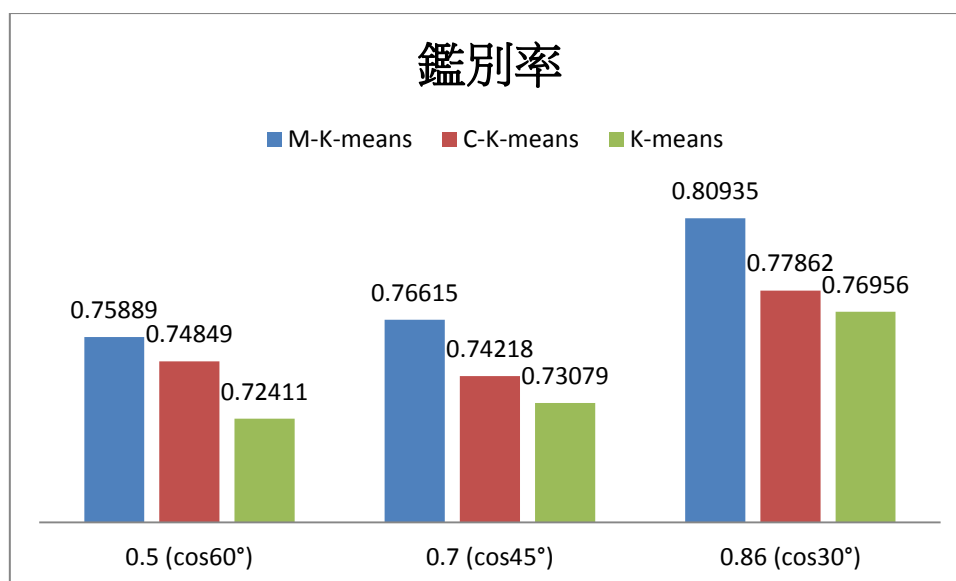


圖 28 鑑別率

由上述的結果可以觀察出改良式 K-means 分群的鑑別率，比其他兩種分群方法好，代表對本篇論文使用的資料集而言，本篇論文提出改良式 K-means 分群演算法，可以有效地分出不相似的群集，提高群與群之間的差異性，對使用者來說能更有效地分析出所需要的資料。

4.3 推薦系統評估

評估推薦系統的評估的方法本篇論文使用 RMSE(Root Mean Square Error)平均絕對誤差值作為評估推薦系統的標準，本篇論文選擇 RMSE 是因為本篇論文的資料來源 Yahoo 與 KDD CUP 聯合舉辦的音樂推薦系統比賽[7]所使用的評估方法也是使用 RMSE，所以本篇也使用 RMSE 做為評估音樂推薦系統的標準，RMSE 的公式如公式 4.3 所示：

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (X_i - P_i)^2} \quad (4.3)$$

X 為音樂推薦系統產生出的預測評分，P 為使用者實際的音樂評分，評估的結果越小越好。本實驗將以傳統 K-means 分群、改良式 K-means 分群、

CURE-K-means 三種分群進行雲端音樂推薦系統的推薦預測，產生的預測評分進行 RMSE 的評估計算，如圖 29 所示：

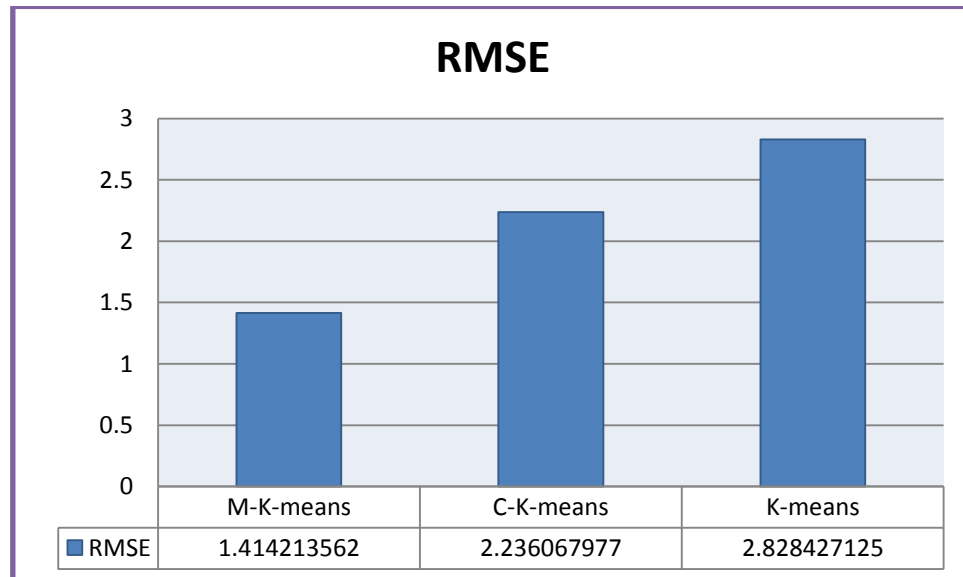


圖 29 RMSE

經由 RMSE 的評估結果，發現使用改良式 K-means 分群演算法作為雲端音樂推薦系統的輸入模組，推薦的結果比使用 CURE-K-means、K-means 的結果還要好，這是因為對於搜尋相似的使用者來說改良式 K-means 分群演算法少了變數會影響分群結果的變量，取而代之的是找出可以囊括全部資料點的空間進行切割作為初始群中心，以穩定隨機初始群中心造成的分群不佳結果，再判斷群內所有資料點與群中心間的相似度，如果相似度低再分群，以提升群內的相似度，推薦系統模組使用經過媒合相似度高的使用者進行推薦，提升推薦的準確度。

4.4 系統展示

本節將展示以移動式手持裝置搭載 Android 系統連結雲端音樂推薦系統的操作畫面，一開始使用者從移動式手持裝置執行雲端音樂推薦系統的服務程式，使用者將會看到系統首頁畫面如圖 30 所示，系統首頁提供兩種推薦方式，分別是個人化推薦與個性化推薦，個人化推薦就是使用者在雲端音樂推薦系統裡已有使用評分紀錄，系統將會依照使用者的喜好進行分析評估，所以使用者只需輸入自

己的帳號，系統便會產生符合使用者喜好的音樂推薦清單。

個性化推薦是以雲端音樂推薦系統裡所有評分紀錄的使用者為背景，當一位新使用者需要雲端音樂推薦系統的推薦服務，但是系統裡並沒有該使用者的評分紀錄，系統便無法產生符合使用者音樂喜好的音樂推薦清單，所以系統便提供使用者一個輸入介面，讓使用者對自己的音樂喜好進行評分，使用者將評分結果送出，系統便會分析出擁有評分紀錄的使用者哪一位的音樂喜好是最接近使用者(被推薦者)，系統便會產生符合使用者喜好的音樂推薦清單推薦給使用者。



圖 30 雲端音樂推薦系統首頁

4.4.1 個人化推薦

使用者登入自己的帳號，如圖 31 所示，雲端音樂推薦系統將會收到使用者的登入訊息，並判斷該使用者與資料庫中的使用者是否相符，與資料庫的媒合成

功，雲端音樂推薦系統便會將推薦結果回傳至移動式手持裝置如圖 32 所示，移動式手持裝置收到雲端音樂推薦系統以 JSON 資料交換格式傳回的推薦結果，移動式手持裝置便會解析資料內容，結果將會以 ListView 的呈現方式呈現在移動式手持裝置的畫面上如圖 33 所示，使用者可以參考雲端音樂推薦系統所帶來的結果，讓使用者可以聆聽到更多不同的音樂。



圖 31 個人化推薦帳號登入

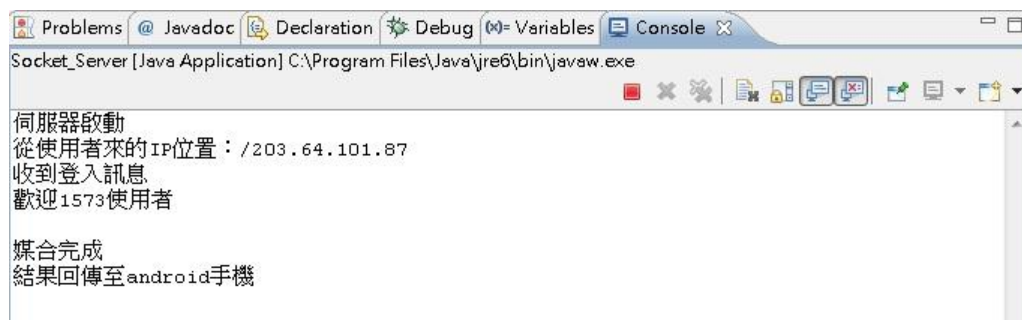


圖 32 回傳推薦結果



圖 33 個人化推薦結果

4.4.2 個性化推薦

使用者選擇個性化推薦，如圖 34 所示，雲端音樂推薦系統將帶使用者進入音樂曲風喜好的評分，使用者可以對自己感興趣的音樂曲風進行 0~100 的評分如圖 35，系統會進行分析並判斷擁有評分記錄使用者的音樂喜好，那一位的喜好與使用者最為相似，與資料庫的媒合成功，雲端音樂推薦系統便會將推薦結果回傳至移動式手持裝置，並將結果呈現在移動式手持裝置的畫面上如圖 36 所示，使用者可以參考雲端音樂推薦系統所帶來的結果，讓對於沒有評分紀錄的使用者也可以聆聽到更多不同的音樂。



圖 34 個性化推薦



圖 35 個人音樂曲風評分



圖 36 個性化推薦結果

第五章 結論與未來展望

5.1 研究結論

本篇論文以音樂作為推薦項目，分析了推薦系統對現今流行音樂市場的幫助，對於現今累積數量龐大的音樂市場與大量的音樂資訊，可能淺藏著使用者感興趣的音樂；卻因數量過於龐大而沒有接觸機會的音樂提供給使用者一個可靠的參考，可以快速幫助使用者找到喜愛聆聽的音樂項目。

使用本篇論文提出的改良式 K-means 分群演算法進行使用者對於音樂喜好的相似度分群，找出相似的使用者進行媒合匹配，分群的結果分出對於不同音樂喜好的使用者群集，使用結合 Android 系統的手持式移動裝置與雲端音樂推薦系統連結，登入使用者 ID 系統便會進行匹配找出與使用者相似喜好的群集，以使用者喜好相似的所屬群集為基礎進行協同式過濾技術的計算，參考該群集的所有音樂項目作為參考推薦給使用者的音樂項目。推薦系統模組將對推薦給使用者的音樂項目進行預測的評分，產生預測的評分方法先使用皮爾森相關係數進行計算；考量被推薦者、推薦者、音樂項目三者的關係進行衡量，產生的相關係數作為預測評分模組的變數進行計算，產生出的預測分數推薦系統模組以 Top-N 方式將預測分數高的推薦給使用者，最後雲端音樂推薦系統會將推薦的結果傳送至手持式移動裝置上呈現給使用者。

實驗結果顯示以改良式 K-means 分群演算法進行推薦，推薦的結果比使用 CURE-K-means、K-means，根據 RMSE 的評估結果顯示改良式 K-means 分群為：1.414213562、CURE-K-means 為：2.236067977、K-means 為：2.828427125，CURE-K-means 演算法以改良式 K-means 分群演算法實現改善率為：36%，K-means 演算法以改良式 K-means 分群演算法實現改善率為：50%，結果顯示使用改良式 K-means 分群演算法進行推薦比使用傳統 K-means 演算法的推薦結果；推薦準確度增加了 50%，實驗結果證明改良式 K-means 分群演算法能幫助推薦

系統提高推薦的準確度，提升推薦的參考價值。

5.2 未來展望

本篇論文提出改良式 K-means 分群演算法改良傳統 K-means 分群的缺點並提升推薦系統的推薦準確度，針對本篇論文的推薦系統未來可進行的工作有：

- 提高推薦的效能，本篇論文使用的資料庫使用者對音樂的評分數量相當多，對推薦模組來說參考的資料相當多，但運算量提高，降低推薦的效能，未來可以針對硬體的配置或改良演算法，以提高推薦的效能。
- 本篇論文以分割式分群演算法找尋相似的使用者，但先前的文獻指出有許多演算法對於搜尋相似使用者有許多不同的作法，可以嘗試使用其他的演算法找出其他相似的使用者，提高使用者間的相似率增加推薦的效能。



參考文獻

- [1] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, 2001, “Item-based collaborative filtering recommendation algorithms”, In Proceedings of the 10th International WWW Conference, pp. 285-295.
- [2] J. MacQueen, 1967, “Some Methods for Classification and Analysis of Multivariate Observations”, In Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability, Vol. 1, pp. 281-297.
- [3] 陳昭宇, 2005, “根基於自我組織特徵映射圖為基礎之最佳化演算法之推薦系統”, 國立中央大學資訊工程研究所。
- [4] J. S. Breese, D. Heckerman, and C. Kadie, 1998, “Empirical Analysis of Predictive Algorithms for Collaborative Filtering”, in Proceedings of the Fourteenth Annual Conference on Uncertainty in Artificial Intelligence (UAI-98), pp.43-52.
- [5] 維基百科－k-nearest neighbor algorithm： [http://en.wikipedia.org/wiki/Nearest_neighbor_\(pattern_recognition\)](http://en.wikipedia.org/wiki/Nearest_neighbor_(pattern_recognition)).
- [6] Q. Li, and B. M. Kim, 2003, “Clustering Approach for Hybrid Recommender System”, in Proc. of the IEEE/WIC International Conference on Web Intelligence, pp.33-38.
- [7] KDD CUP from yahoo lab, 2011, <http://kddcup.yahoo.com/>.
- [8] Netcraft, 2011, <http://news.netcraft.com/>.
- [9] 維基百科－長尾： <http://zh.wikipedia.org/wiki/長尾>.
- [10] A. Chirs, 2006, “The Long Tail: Why the Future of Business Is Selling Less of More”, Hyperion Books New York.
- [11] Z. Zaier, R. Godin, and L. Faucher, 2008, “Evaluating Recommender Systems”,

- Automated solutions for Cross Media Content and Multi-channel Distribution on International Conference, pp. 211-217, Nov.
- [12] U. Shardanand, and P. Maes, 1995, "Social Information Filtering: Algorithms for Automating 'Word of Mouth'", in Proceedings of the Conference on Human Factors in Computing Systems (CHI95), pp. 210-217.
- [13] 行政院新聞局－臺灣流行音樂產業調查：
<http://www.gio.gov.tw/publish/2010music/>
- [14] M. B. AI-Daoud, and S. A. Roberts, "New methods for the initialisation of clusters", ELSEVIER, Pattern Recognition Letters, pp. 451-455.
- [15] 許榮望, 2009, "推薦系統的研究", 吳鳳技術學院資訊管理系。
- [16] B. Chen, P.C. Tai, R. Harrison, and Y.. Pan, 2005, "Novel Hybrid Hierarchical-K-means Clustering Method (H-K-means) for Microarray Analysis", In Proceedings of 2005 Computational Systems Bioinformatics Conference, Workshops and Poster Abstracts. IEEE, pp. 105-108.
- [17] D.M. Kim, K.S. Kim, K.H. Park, J.H. Lee, and K.M. Lee, 2007, "A Music Recommendation System with a Dynamic K-means Clustering Algorithm", Machine Learning and Application on International Conference, pp. 13-15, December.
- [18] L. I. Kuncheva, 1995, "Editing for the k-nearest neighbors rule by a genetic algorithm", ELSEVIER, Pattern Recognition Letters, pp. 809-814.
- [19] W. G. Teng*, H. H. lee, 2007, "Collaborative Recommendation with Multi-Criteria Ratings", Journal of Computers (Special Issue on Data Mining), pp. 69-78, January.
- [20] M. N. Uddin, J. Shrestha, and G.S. JoEnhanced, 2009, "Enhanced Content-based Filtering using Diverse Collaborative Prediction for Movie Recommendation", Intelligent Information and Database System on First Asian Conference, pp.

132-137, September.

- [21] R. Ng, and J. Han, 1994, "Efficient and Effective Clustering Method for Spatial Data Mining", In Proc. 1994 Int. Conf. Very Large Databases (VLDB'94), pp. 144-155.
- [22] 林育臣, 2002, "群聚技術之研究", 朝陽科技大學資訊管理研究所。
- [23] 陳榮昌, "群聚演算法及群聚參數的分析與探討", 朝陽科技大學資訊管理研究所。
- [24] S. Guha, R. Rastogi, and K. Shim, 1998, "CURE: An efficient clustering algorithm for large databases", In Proc. 1998 ACM-SIGMOD Int. Conf. Management of Data (SIGMOD'98), pp. 73-84, June.
- [25] S. Guha, R. Rastogi, and K. Shim, 1999, "ROCK: A Robust Clustering Algorithm For Categorical Attribute", In Proc. 1999 Int. Conf. Data Engineering (ICDE'99), pp. 512-521, March.
- [26] G. Karypis, E.H. Han, and V. Kumar, 1999, "CHAMELEON: Hierarchical Clustering Using Dynamic Modeling", IEEE Computer, Vol. 32, No. 8, pp. 68-75.
- [27] V. Estivill-Castro, and I. Lee, 2000, "AMOEBA: Hierarchical Clustering Based on Spatial Proximity Using Delaunay Diagram", In Proc. 9th Int. Spatial Data Handling (SDH2000), pp. 10-12, August.
- [28] T. Zhang, R. Ramakrishnan, and M. Livny, 1996, "BIRCH: An Efficient Data Clustering Method for Very Large Databases", In Proc. 1996 ACM-SIGMOD Int. Conf. Management of Data (SIGMOD'96), pp. 103-114.
- [29] 陳同孝, 2006, "結合K-means及階層式分群法之二階段分群演算法", 國立臺中技術學院資訊科技與應用研究所。
- [30] M. Ester, H.P. Kriegel, J. Sander, and X. Xu, 1996, "Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise", In Proc. 1996

Int. Conf. Knowledge Discovery and Data Mining (KDD'96), pp. 226-231, August.

- [31] W. Wang, J. Yang, and R. Muntz, 1997, "STING: A Statistical Information grid Approach to Spatial Data Mining", In Proc. 1997 Int. Conf. Very Large Data Bases(VLDB'97), pp. 186-195, August.
- [32] D.H. Fisher, 1987, "Knowledge Acquisition Via Incremental Conceptual Clustering", Machine Learning 2, pp. 139-172.
- [33] L. Kaufman, and PJ Rousseeuw, 1990, "Finding Groups in Data: an Introduction to Cluster Analysis", John Wiley & Sons.
- [34] M. Ankerst, M. Breuning, H.p. Kriegel, and J. Sander, 1999, "OPTICS: Ordering points to identify the clustering structure", In Proc. 1999 ACM-SIGMOD Int. Conf. Management of Data (SIGMOD'99), Philadelphia, PA, pp. 94-105, June.
- [35] R. Agrawal, J. Gehrke, D. Gunopulos, and P. Raghavan, 1998, "Automatic Subspace Clustering of High Dimensional Data for Data Mining Applications", Int. Conf. Management of Data, pp. 94-105.