

# 以時間參數為基礎之智慧型音樂 播放清單推薦系統設計

廖偲惟

國立屏東科技大學

資訊管理系

研究生

m9556012@mail.npust.edu.tw

簡翡瑢

國立屏東科技大學

資訊管理系

研究生

m9656028@mail.npust.edu.tw

劉寧漢\*

國立屏東科技大學

資訊管理系

助理教授

gregliu@mail.npust.edu.tw

## 摘要

現今網路快速發展下，帶動了數位音樂的普及，使用者漸漸能藉助網路隨時聆聽到他們所喜好的音樂歌曲，更在電子商務引領下發展出推薦系統，提高使用者對音樂購買的慾望。目前一般線上音樂推薦系統，經常擷取從過去的歷史紀錄，以進行資料分析或透過統計方式來推薦其它人常聽的歌曲，但往往使用者的需求不僅如此，因為使用者經常受時間或地點等因素，而選擇或改變聽歌的行為，若系統只考慮到使用者喜好的歌曲類型，似乎對個人化的推薦服務仍不夠完整健全。所以本研究於音樂播放清單中欲加入時間排程的概念，並結合決策樹分類技術，期望推薦更符合使用者當前適合的音樂。最後，經由實作與分析後推薦準確度的結果，已達到我們所預期的目標成效。

**關鍵詞：**音樂推薦系統、決策樹、時間連續性、內容過濾、合作式過濾。

## Abstract

The digital music is booming through rapid expansion of Internet. Users can listen to their favorite songs via web all the time. The electronic commercial leads to development of the recommendation system, which enhances the desire of music buying for customers. The online music recommendation system usually grabs the historical record from past listeners. With extensive data analysis or statistical means, the system recommends the popular music to others. However, users' demand is far beyond that because their choice or change of listening behaviors is often affected by various factors, such as time or place. If the system only considers the type of favorite songs for users, it seems not a comprehensive service for personal recommendation system. So this research will add time scheduling to the music playlist, and combines classification technology of decision tree to suggest users the suit music more

precisely. Eventually, the accuracy of recommend results achieved in our anticipate result after implementation and analysis.

**Keywords :** music recommend system、decision tree、temporal continuity、content-filtering、collaborative-filtering

## 1.前言

隨著線上的音樂市場規模不斷擴展下，音樂類型也更具多樣性，往往使用者在搜尋音樂資訊時，需耗費許多的精神與時間在收集方面。在此情況下，系統也常會因為缺乏使用者的喜好資訊，而無法產生個人化需求的窘境。有鑑於此，許多研究學者便紛紛投入於個人化推薦服務的行列，並在網路快速促擁下，使用者能藉由線上音樂系統隨時登入聆聽，並記錄使用者聽過的歌曲，但傳統音樂推薦系統只能產生每週或每月點閱率的排行，這已漸漸不敷使用者個人的需求。

所以在數位音樂興盛的同時，也慢慢發展出不同的個人化推薦服務與技術，在處理龐大資料時，會運用過濾系統(Filtering System)，來幫助我們篩選出所需求的訊息，例如：「潘朵拉」[30]，它是最典型以音樂內容來達到音樂推薦的目的，主要是藉助多種音樂特徵，分成不同的類別，再將具有相似特徵的音樂，推薦給有相同興趣的使用者，而其它像「iLike」[16]的音樂推薦系統，主要區別在於使用者可以在網站的歌手庫中選出自己本身的喜愛歌手，通過這個喜好的選擇，系統能幫助使用者找出關聯喜好的同好使用者，以實現同好的音樂社群(Social network service,SNS)，不同於iLike的音樂推薦系統像是「Last.fm」[22]，則是透過使用者所點播過的音樂，去計算使用者間的相似程度，便可找出同好相似群作出推薦，由此可見，推薦服務已然成為時勢所趨。而我們應如何更準確推薦使用者當前需求的喜好音樂，以然成為我們研究推薦系統首要的考量重點。

\*通訊作者

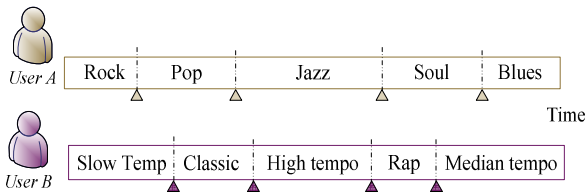


圖 1 各時間區段的行為表現(▲:時間分割點)

儘管眾多的音樂推薦系統已解決多數關於個人化推薦的問題，但因為使用者本身的喜好行為，可能隨著時間產生變化，所以時間因素的考量值得我們去研究探討。為了讓推薦能更具人性化且貼近使用者生活習性，故本研究提出於播放清單中加入時間排程的概念，並結合內容過濾(Content-Filtering)與合作式過濾(Collaborative-Filtering)之混合推薦機制於音樂推薦系統，由使用者的回饋(Feedback)，透過代理人學習使用者喜好行為，並由系統自動調整及安排個人化播放清單，以取得更適合於使用者當前的音樂資訊，我們以圖 1 顯示兩個不同使用者於時間軸上的喜好差異表現。

圖中代表時間點分割後結果，能明顯呈現使用者在每個時段所偏好的類型音樂會有不同。倘若考慮的時間切割位置改變時，其時間區間範圍內資料大小也同樣產生改變，這可能影響到分類模型的結果與準確性，所以時間分割點的切割位置，須具有其時間上的意義，才能為個人於時間區段作出正確的決策模型。所以本研究為此找到一個適當的演算法以解決時間切割點問題，期望讓分割後所產生的決策樹模型能表現當前時段使用者的喜好行為，並經由模型建置產生喜好程度上的差異，進而安排適時地個人化播放清單，透過時間概念的加入，以提昇在推薦準確性的目標。

本論文其它章節如下，在第 2 節我們針對目前推薦系統的類型作描述與相關研究的探討；第 3 節介紹本研究所要解決的問題以及我們所提出的策略方法；第 4 節則是我們的實驗以及說明討論；最後，第 5 節為結論與未來研究方向。

## 2. 相關研究

### 2.1 推薦系統

推薦系統(Recommend system)是一種可以有效篩選資訊，提供個人化服務的工具，並且在大量可能的選擇項目中，為使用者篩選出最感興趣的資訊，此為我們發展推薦系統的首要宗旨。基本上，推薦系統主要可根據使用者對項目的喜好评比、內容相似性或是瀏覽行為等三面向為出發點。我們就目前常見音樂推薦系

統類型，根據其演算法的不同，將它大致分為內容導向式的推薦方法(Content-Based,CB)、合作式過濾之推薦方法(Collaborative-Filtering, CF)、混合式推薦法(Hybrid Recommendation Method)、「情境推薦法」等四大類。

#### 2.1.1 內容導向式

國內外學者的研究中，已有許多研究關於數位音樂，有的是針對音樂的外部特徵為出發點來進行音樂分類，並提出以音樂的各種元數據(Metadata)來描述音樂的外部特徵，藉此達到音樂訊息的有效管理；而另有學者也提出從音樂內容著手研究，主要是從音樂擷取其中內容特徵來進行分析、預測及推薦，根據項目間的關聯性，找出項目的屬性特徵及使用者歷史資料喜好加以分析，進而提供推薦結果，稱為基於內容導向式(Content-Based,CB)。如 [23][35][41] 都是從使用者過去的偏好資料中，分類出使用者有興趣的內容特徵，而當萃取出類似的項目，則進而推薦給使用者，在一般音樂的推薦系統中，首先會為使用者建立特性輪廓(User Profile)，此 User Profile 中主要是紀錄一些關鍵詞及進一步紀錄使用者相關資訊。經由過濾系統為每新加入的歌曲與音樂內容作資料分類、註解、比對的動作，以保證過濾後的內容和用戶的興趣相吻合，如：音樂特徵、Meta Data 等，此為內容導向過濾推薦技術(Content-based Recommender)，著名的系統有 NewsWeeder [17]。另有學者 Daniel McEnnis [7] 思考因文化背景所影響的音樂歌詞，描述出有種族衝突或反應社會現象，可藉由歌詞特徵分析，找到相同喜好的使用者；Steffen Pauws & Sander van [34] 是根據 user profile 及回饋評價，利用音樂內容特徵作分析基礎，來產生播放清單，其基本內容分析如下圖 2 所示。

基於內容推薦方法的優點：1. 透過推薦項目的內容特徵，藉此能分類項目並推薦相似的歌曲；2. 能推薦給具有特殊興趣偏好的使用者；3. 能推薦新的或未聽過的歌曲，以解決推薦新歌曲的問題；4. 不需其他使用者的資料即可作推薦，解決 Cold-start 和資料稀疏問題。

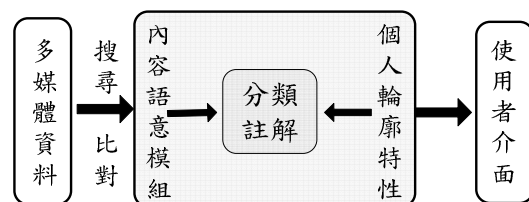


圖 2 基本內容式的資料過濾

其缺點在於使用者所喜好音樂內容特徵無法被顯示。

### 2.1.2 合作式過濾法

由Goldberg et al.[11]首先提出，此方法也稱作社會過濾(Social Filtering)方法，使用者不須處理其中的項目內容或瞭解其中知識(如：音樂內容特徵分析)，只須透過使用者過去的音樂喜好，計算出興趣相似的使用者來形成社群，並以目標相近使用者對項目的評價另作加權計算，進一步預測使用者對項目的喜好程度，此為合作式過濾的目標，使用此方法的音樂推薦系統，如Ringo[36]主要是藉由收集使用者對歌曲的評價後，系統能透過這些的評價資料，以合作式推薦方法，由電子郵件(mail)作發送推薦，之後當使用者增加其喜好歌手的評比時，則會隨使用者興趣去改變推薦清單；另一傳統的音樂推薦系統，如Amazon.com[1]則是藉由信賴的朋友(FOAF)構成網路社群，並根據FOAF (friend of a friend)的同好進行互相推薦。與Amazon.com 相似的還有Oscar Celm [28]，藉由每個使用者輪廓取得個人喜好後，再結合RSS 的新聞快報，擷取出相關使用者的喜好資訊(如：歌手名稱、歌曲、音樂類型)，作為FOAF的過濾，藉此可以由新聞(RSS)適時取得新的歌曲或歌手相關資訊以便作為未來的推薦清單。合作式過濾推薦技術，其主要可大致分為兩大類：一種是以記憶為基礎的合作式過濾(Memory-based Collaborative Filtering, MB-CF)：透過統計計算方式，將具有相似偏好的鄰居使用者分群，此方法又稱合作式過濾(User-based Collaborative Filtering, UB-CF)或稱鄰居合作式過濾(Neighbor-based Collaborative Filtering, NB-CF)，另一派學者提出以模型為基礎的合作式過濾(Model-based Collaborative Filtering)：就使用者過去的歷史資料建立使用者模型(User Modeling)，透過此模型可為使用者進行預測推薦，且此模型的推薦技術已廣泛應用：包含有貝式網路(Bayesian Networks)和潛在語義檢索(Latent Semantic Indexing)、類神經網路(Artificial Neural Network, ANN)學習、決策樹(Decision Tree)分類等技術，專為訓練樣本來建構模型。

合作式過濾主要是找出使用者與使用者偏好的音樂項目的關聯性(如：利用評價或標籤)，系統除大量紀錄使用者輪廓(User Profile)，如記錄使用者的個人興趣、背景、知識等，並以統計計算使用者之間的相似度，進而找出具有相似喜好的鄰居(Neighborhood)，藉

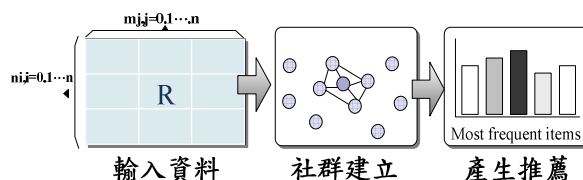


圖3 使用者合作式過濾分群過程

由得知他們相似的喜好，來進行預測及推薦，此方法即稱為合作式過濾，亦是推薦技術最廣為應用的推薦方式，目前已有很多技術都是圍繞合作式過濾發展研究，如[4][15][17][18]這些的研究皆以此方法加以應用，基本上都以使用者的經常訪問或是從瀏覽行為中得知其喜好，以找出其它同好使用者，並以此近似喜好的資源作為推薦目標，另外Y.S.Tzeng[42]則根據使用者的評價從中找出相同興趣的社群成員，以聽過的歌曲來作為推薦的項目。學者Sarwar認為合作式的推薦處理過程主要可包含三步驟，分別為輸入資料、社群的建立和產生推薦，如圖3所示。

- 輸入資料表示法：將使用者的過去行為及興趣利用一個  $m \times n$  的矩陣並以  $R$  來表示，即  $n$  個使用者喜好  $m$  個項目的歷史資料，矩陣元素  $R_{ij}$  則表示第  $i$  個使用者喜歡第  $j$  個項目。
- 相似社群建立：首先，是透過合作式過濾，由統計計算出使用者間的相似程度，作為未來推薦的依據。
- 產生推薦：爾後，從社群用戶中進一步得知目標使用者在前  $n$  筆項目裡最常被選擇的項目，以作為首要的推薦。

合作式推薦具有以下優點：1.能夠過濾不易進行自動內容分析的資料數據，如音樂或多媒體資料等...；2.避免內容分析的不精確，可以透過其他人的經驗，對一些複雜難以表述的概念(如：個人喜好的歌曲內容特徵)進行過濾；3.具有推薦新項目的能力，用戶對推薦的項目可能會有意想不到的效果產生，這也是合作式過濾和基於內容導向過濾最大的差別所在，因為合作式過濾可以發現自己尚未發現的喜好興趣，即使合作式過濾的方法是已被廣為應用的，但仍有許多的問題(Cold-start、資料稀疏...等)問題，導致初始系統推薦效果較差，也有待各學者去解決。

### 2.1.3 混合式推薦法

由於各種推薦方法都有其優缺點，便有學者提出將內容導向式和合作式過濾法作組合，產生混合推薦法(Hybrid Recommendation Method)，如此能夠適當解決上述推薦的缺點(Claypool et al.,1999)，以產生最佳推薦預測結

果。學者Beth Logan.[3]、Hung-Chen Chen[12]皆採以使用者的profiles為基礎，將過去的喜好資料輔以音樂內容特徵作分類，找出相近使用者喜好的音樂歌曲後，利用使用者所喜歡的類型作為推薦，並透過計算使用者的profiles間的相似度，進一步產生相同群組使用者的推薦清單。另有學者Wang, H.[41]、Claudiu S[6]其系統著重在使用者所標註的標籤(Tag)，以作為使用者合作式分群之依據，當有其它使用者也同樣下過此標籤時即可視為相同用戶群，主要可以由初步分群來減少時間的計算，之後再根據音樂內容來作進一步分析，以達到互相推薦的目標。

像其它學者Kuo, F.-F.[20]、Shan, M.-K[34]等也都是採此種混合式推薦來運用於音樂推薦系統上，以實現個人化的推薦。綜合來說，混合推薦方式的主要策略為：

- 1.加權(Weight)：加權多種推薦技術結果。
- 2.變換(Switch)：根據問題背景與實際情況，改變所採用的推薦技術。
- 3.混合(Mixed)：同時採用多種推薦技術給出多種推薦結果為使用者提供參考。
- 4.特徵組合(Features combination)：組合來自不同的資料特徵，為另一種推薦演算法所採用。
- 5.層疊(Cascade)：先用一種推薦技術產生一種粗糙的推薦結果，第二種推薦技術再產生更精確的推薦結果。
- 6.特徵擴充(Features augmentation)：一種技術產生附加的特徵資訊，嵌入到另一種推薦技術的特徵輸入中。

上述三類的推薦方法是經常被音樂推薦系統所使用，針對此三類方法的優缺點，整理如表 1 所示。

表 1 推薦方法之比較

推薦方法	優點	缺點
內容式過濾推薦	可以推薦使用者特殊喜好或有興趣的音樂	1. 複雜屬性不好處理 2. 需足夠資料才能分類
合作式過濾推薦	1. 不需要太多的領域知識 2. 隨著時間推移性能提高處理複雜的非結構化對象 3. 能推薦未見過之新項目	1. 稀疏問題：評比數太少 2. 新使用者問題 3. 系統初始時，推薦準確性較差
混合式推薦	結合各種過濾優點，用以解決特定問題。	需解決所制定的特定問題

總結來看，之前的音樂推薦系統，大都是採類似內容式或合作式資訊過濾的技術，並由使用者輪廓(如：喜好的歌手或歌曲、或從音樂內容特徵、音樂文化背景...等)方面著手，也有結合在其它技術，例如：RSS、標籤、語意描述或以人工智慧建立使用者輪廓模型來提高推薦的準確度，但在個人化的音樂推薦上顯少考慮到環境因素的重要性，此是不容被忽略的需求，因此即有學者提出情境因素的分析，以找出適合某些情境下使用者有興趣的項目。

#### 2.1.4 情境推薦法

通常在選擇要聽歌曲的同時，使用者勢必會考慮當下的情況而作出選擇，例如：在某個地點或個人情緒，選擇想聽什麼類型的歌曲，此想法即是考慮推薦的情境資訊Herlocker & Konstan [13]，此類方法亦能因每個使用者所處情境的不同，推薦適合的音樂給使用者聆聽。如Katsuhiko Kaji[19]即是利用使用者對喜好音樂清單加上標籤註解(如：感覺、情緒表達、地點描述)，藉由標籤註解，透過三個特徵空間的分析(歌詞、音樂背景、收聽地點等)分別與其他相似喜好的User作群集，藉由相似使用者間互相推薦喜好的歌曲，以達情境推薦效果。

現實環境中，使用者有時聽歌會受不同情境所牽引。因此本研究的推薦系統欲強調在個人喜好行為與時間上的緊密關係，由於時間屬於連續性資料的情況下，我們要如何找到一個有效處理方式，成為我們研究首要解決的問題。另外因為使用者的喜好是會改變的，所以我們還須為本系統設計一個良好分類學習機制，即針對個人的喜好資料與回饋紀錄加以修正及調整，以達到更準確推薦的目的。

#### 2.2 播放清單產生器

音樂推薦系統拖離不了播放清單產生(Playlist Generation)方式，一般的播放清單產生器較為傳統刻板，常見的方式是以隨機拖拉或手動點選方式產生，而Rob van Gulik & Fabio Vignoli[33]學者則是採以視覺化技術(visualization technique)方式來呈現，由使用者任由在地圖中點選某一點(point)，根據屬性標籤(如：mood、genre、year、tempo)，將自動於地圖上找到相對應的位置，此位置範圍內可輕易看見其它相似的歌曲，這不僅能讓使用者能更容易發現新的且未聽過的類似音樂，而且此種方式對使用者來說也增添與系統之間的互動性。則有學者Elias Pampalk[8]，是根據系統產生的啟始音樂，讓使用者立即作出回饋，此回饋則是根據使用者在播放清單上的動作行

為(如: skip), 當按下skip按鈕則表示與當前的使用者喜好不符, 如總是跳過某些特定歌手, 系統則會將此回饋當作下次推薦的依據, 再由音樂內容的相似性來找到符合的內容加以推薦。另有Arthur Flexer[2], 學者是透過介面由使用者決定輸入的啟始音樂與結束音樂, 再以音訊相似找出兩者類型間的漸層關係後, 並採用漸層概念來安排歌曲清單的呈現方式, 即可自動為使用者安排啟始與結束之間的歌曲清單, 而不須用到其它標籤資料, 透過這樣的方式不僅間接能夠讓使用者發現其它不同多元的類型音樂, 更達到個人化清單產生的目的。近年更有學者Nuria Oliver[27]提出PAPA的方法(Physiology and Purpose-Aware Automatic Playlist, PAPA), 此方法特別的應用, 是針對使用者在運動期間, 根據使用者的生理反應與Purpose-Aware來安排即時地個人化音樂播放清單, 當運動者步伐加快時, 則可能播放慢歌; 反之, 若運動者步伐變慢時, 則清單即時在下首調整安排較快的音樂, 試圖幫助調節運動者的步伐與心律節奏, 這些播放清單的產生總脫離不了對於音樂資料的分類處理, 在下節我們將對於分類學習法作進一步探討。

## 2.3 分類學習法

分類學習法(Classification Learning)是一種常被應用在機器學習領域的一種學習方法, 資料探勘中決策樹(Decision Tree)已是常被用到的其中一種分類學習技術, 一般在資料分析或預測方面, 會透過每個節點中資料集作測試, 來達到分類分析的目的, 所謂分類(Classification)即是根據已知的類別屬性, 對資料集來建立分類模型, 並可利用它來預測新進資料或未知資料的類別, 當中相當有名的一種分類學習法, 為Quinlan在1986年所提出ID3分類樹演算法, 此演算法在分類樹的應用上已有相當不錯的表現, 在1993年時更針對ID3無法處理連續性數值的缺點上作進一步地改進, 提出C4.5分類學習演算法, 下面我們將各別針對ID3分類學習法與C4.5分類學習法詳加介紹。

### 2.3.1 ID3 分類學習法

ID3 分類學習法, 因為能產生精簡的決策樹, 且在計算空間成本均小情況下, 已被廣泛運用在許多領域Fu[9]。但由於ID3只適用在處理屬性值項數較少的名目屬性(Nominal Attributes), 而不易處理帶有連續數值屬性的資料集Fayyad and Irani[10], 尤其當同時存在有

連續數值屬性與多值的資料集的應用, 往往造成節點與分支過多, 降低分類能力。ID3的核心演算法, 主要以資訊增益(Information Gain)作為選擇最佳屬性的度量標準, 在決策樹各級節點上選擇屬性時, 其具體步驟是一開始利用資訊熵函數(Entropy function)計算於所有各屬性資料集, 之後優先選擇Information Gain最大的屬性產生決策樹的根節點, 再由該屬性中的子集來建立分支, 針對各分支的子集遞迴調用Entropy function方法, 來建立其它決策樹節點的分支, 最後可得到一棵分類樹, 便可利用此分類模型來對新的樣本資料進行決策。

其ID3分類演算法的優點: 演算法的理論清晰且方法簡單, 學習能力強; 其缺點: 若屬性中帶有多值或連續數值, 會傾向劃分較小的類別甚至單一類別, 且對雜訊較敏感。如被分割後的子集都只有一個資料, 則亂度為零, 這會造成樹被建出後分支過大, 且容易造成過度學習的問題發生。在此問題下, Quinlan又提出C4.5進一步改善ID3的不足與缺點。

### 2.3.2 C4.5 分類學習法

C4.5分類學習法改良自ID3而來。主要以資料為導向, 當測試一個連續數值屬性時, 是屬於「由上而下的決策樹歸納法」Quinlan[31], 在建構決策樹時, 是採用監督式作訓練學習, 並由增益率(Gain Ratio)分岔準則來挑選最佳屬性, 之後以二元分割方式建出決策樹根節點(Root node)及內部節點(Internal nodes), 為避免資料中含有雜訊或有資料不完整情況, 在建完C4.5決策樹後會對產生的樹進行分支修剪(Prune), 修剪時會藉由預估錯誤率的計算評斷每個子樹節點與葉節點的錯誤率。修剪後的樹, 可得到最精簡且錯誤率又不會過高的決策模型。

C4.5分類演算法繼承了ID3演算法的優點, 在以下幾個方面對ID3演算法進行改進:

1. 改以Gain Ratio來選擇最佳屬性, 克服了用資訊增益(Information Gain)選擇屬性時, 偏向選擇取值多的屬性缺點。
2. 能處理離散型及連續性數值屬性。
3. 能夠處理不完整的資料集。

C4.5演算法與其它分類演算法, 如統計方法、類神經網路等方法比較起來。有以下優點: 產生的分類規則易於理解, 準確率較高。其缺點是: 在建構樹的過程中, 需要對資料集進行多次的順序掃描與排序, 因而導致演算法的效率降低。而在搜尋每個分割點時, 可能會

因屬性值過多而在分類後建出的樹過於龐大，以致系統效能降低的問題。

## 2.4 音樂特徵值擷取

在傳統的音樂資料庫中，音樂的分類及查詢不外乎是基於演唱者、曲風、年代等資訊，而這些資訊通常都是以人工的方式取得，因此在大量的音樂資料庫中，這種方式就會相當沒效率，就自動化分類及內涵式查詢法而言，適當的特徵值擷取就相當的重要。從音樂的表示法，可分為以訊號為主的資料，如：WAV、MP3等，及符號為主的資料，如：MIDI兩類。對於訊號而言可以直接取用相關低階參數，如 (i) zero crossing rate (ii) MFCC coefficients等 [21][24][38]，或是轉置為符號資料[5][25][32]。

由於音樂特徵可視為組成音樂中重要元素，我們可透過混合多種的特徵來表達或描繪一首歌曲的意涵，本研究將從音樂組成的四個主要要素 Hammer, Elenor ray and Malcolm S.Cole[14]，包括音質 (Tone Color)、音高 (Pitch)、音長 (Duration)、力度 (Dynamic) 中取得其中 23 種不同維度的特徵值，透過 Echo Nest Analyze API [39]，將上傳的歌曲 (MP3) 分解成片段 (segment)，再從每個 segment 計算其中不同維度的特徵值，包含的 features 如下 M Slaney[26]：

● 從音質要素中取得重要的「音色」及「節奏」特徵值：

1. 音色 (Timbre)：依振動頻率不同的狀態而有所差異，可從波形與時間振動次數關係找到。經由 short-time Fourier transform 後的資料：包含 spectral centroid, spectral flux, time domain zero crossings, low energy, spectral roll-off and MFCCs。
2. 節奏 (Tempo)：從樂曲進行的拍子速度來看，可用來表示節奏的快慢，經由 discrete wavelet transform 後的資料，計算音高資料中相鄰音符的平均時間差可得之，用以描述音樂的節奏特性。
3. 節奏信度 (Tempo Confidence)：估算節奏的信度 (介於 0~1)。
4. 拍子變異數 (Beat Variance)：敲擊規則性或重複的變化。
5. 拍號 (Time Signature)：屬於感知的計算，非作曲家所標註的。
6. tatum：為音樂層次結構中最低層的 tatum 層，計算 tatum 層總音軌的音長。
7. tatum 信度 (Tatum Confidence)：估算 tatum

的信度 (介於 0~1)。

8. NumTatumsPerBeat：計算 tatum 每個敲擊的次數。
9. time Signature Stability：概略估計跨音軌的拍號穩定度。
  - 音高 (Pitch)：為單音符的升降，構成音的高低，屬於音的水平變化。偵測多重音高，將音高利用 pitch histogram 統計，以描述音樂中音高的特性。我們利用 MIDINOTE 擷取特徵工具取出其中相關特徵值。
10. 平均音高 (Mean Pitch)：計算總音軌音高音符的平均數。
11. 音高亂度 (Pitch Entropy)：計算總音高的亂度特徵值。
12. 音高標準差 (Pitch Standard)：計算音高的標準差。
  - 音長 (Duration)：可依振動時間的長短而定，「音長」：表振動的時間長；「音短」：表振動的時間短。同樣以 MIDINOTE 工具，取得其中能描述音樂的音長特性。
13. 平均音長 (Mean Duration)：計算平均的時間長度。
14. 音長亂度 (Duration Entropy)：計算總音長的亂度特徵值。
15. 音長標準差 (Duration Standard)：計算音長的標準差。
16. 規則性音長變異數 (Segment Duration Variance)：有規則性或重複的音長變化。
  - 強度、力度 (Dynamic)：音的強弱，依振動大小而定，振幅越大表音越強。將使用 MIDINOTE 工具，計算出 Volume 值，用以代表每首歌曲的強弱特性。
17. 時間平均最大響度 (Time Loudness Max Mean)：片段中響度最大的平均值或啟奏音長。
18. 音量最大平均 (Loudness Max Mean)：片段中最大響度變化的平均值。
19. 音量最大變異數 (Loudness Max Variance)：片段中最大響度變化。
20. 音量啟始平均 (Loudness Begin Mean)：片段中啟始響度變化的平均值。
21. 音量力度平均 (Loudness Dynamics Mean)：片段中啟始到結束的範圍變化平均值。
22. 音量力度變異數 (Loudness Dynamics Variance)：片段範圍中力度的變化。
23. 音量 (Loudness)：計算總音軌中的響度。

### 3. 系統架構及方法

本研究，在Web環境下發展一套具混合內容導向與合作式的時間排程音樂推薦系統，其主要核心以C4.5[31]決策樹分類學習法為基礎，並以Gain Ratio作最佳屬性的度量。

由於系統在初始推薦時，常遇到資料庫中無新使用者相關的喜好資料，如此無法呈現良好的推薦結果。所以本研究一開始希望能透過合作式過濾機制產生初始清單，適度解決Cold-start的問題。之後，由使用者的回饋與喜好的變動情況進行決策樹模型的建置，最後適時推薦符合當前使用者真正需求，並由代理人透過訓練學習的過程作出最佳決策，之後在本研究實驗中，進行使用者建模並從滿意度資料，分析其中相關現象的原因。

推薦系統核心係以決策樹學習法建立使用者行為模型，主要可分為三個階段，依圖4所示。

□第一階段：收集可代表使用者興趣資料。

首先，在使用者相關喜好方面一系統於推薦初期是缺乏新使用者回饋資料的，因此我們設計由新進使用者填入一般問卷式的問題，先為使用者作初步的喜好調查。使用者可透過線上的推薦系統平台填寫基本職業屬性資料，並勾選個人所愛好的歌曲類型，如搖滾/重金屬、藍調、爵士、古典樂、靈魂樂、流行音樂、華語音樂、西洋音樂等，以作為往後使用者分群依據。

其次，收集訓練樣本方面一經由使用者的回饋得知個人喜好，系統同時將使用者在某些時間選擇的音樂喜好記錄下來，包含標註的時間資料、內容特徵及歌曲評價資料(喜好程度由極喜歡到極不喜歡共分為十等第)，皆經由系統介面予以回應，一併儲存於使用者的瀏覽行為的資料庫當中。系統對於時間的標記處理方式，我們分為以每日時間、日期、星期、季節及假期。例如某首耶誕歌曲在2008年12月25日早上8:00被使用者接受播放，歌曲長度為3分鐘，我們將記錄使用者姓名、樂曲名稱、播放時間8:00 am~8:03 am(標記480 minute)、星期二、25日、冬季及聖誕節，這些時間標記將用作後續處理的重要依據。

□第二階段：合作式推薦模組建置。此合作式的推薦模組將產生初始音樂推薦清單。

首先，依相似使用者分群一在一般使用者聽歌很少情況下，開始時是無從知道要如何選擇的，若是讓使用者評價資料庫中所有的歌

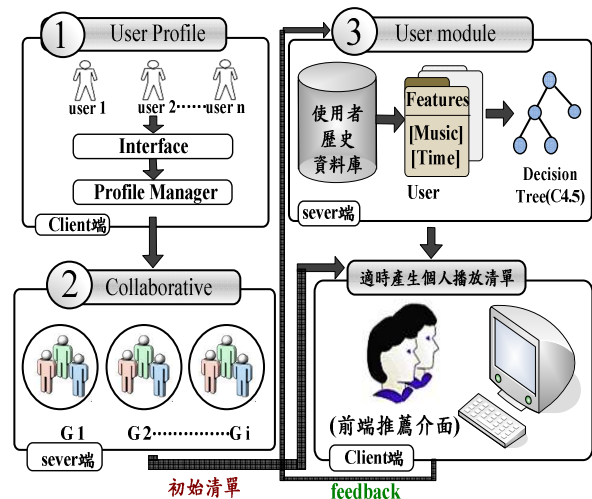


圖4 智慧型音樂排程推薦系統之架構流程

曲，或是當有新歌被建入資料庫中這都可能需經過一段訓練過程後才能把此新歌推薦到清單中，這對於推薦的時效性上顯得沒有效率。為解決此一問題，我們在初始會依據第一階段所收集的相關喜好資料(User Profile)，例如年齡、性別、職稱、喜好的音樂類型等，配合合作過濾法[11]進一步找到相似同好群，在此將為分群法設立一閾值(Threshold value)，當找出K個臨近相似的使用者後，會針對群中使用者過去聽過歌曲的次數給予不同的權值，依喜好程度不同於播放清單中作排序。

其次，產生初始清單一透過此群聚的結果，我們可以進一步利用統計計算權值得到Top-n的歌曲，依序排列並產生初始推薦清單，藉此清單不僅能獲得個人在長期或短期的喜好特性，也能加快正向回饋資料的收集，讓行為模型更順利地被建立。

□第三階段：個人化推薦模組。此模組為本研究主要核心，我們為使用者設計了一個專屬的智慧型音樂排程代理人，由收集到的喜好特徵資料與系統標註的時間資料，經分類學習法建置使用者的行為模型，適時安排目前合適的音樂播放清單，透過不斷地學習與更新使用者的行為模式，當新的歌曲進入到後端資料庫時，系統便能夠透過此一模組對線上的使用者即時比對作出好的決策反應，達到個人化推薦目標。

在我們相較其他的學習演算法中，如類神經網路學習法，雖然在準確度或是容錯能力方面都有不錯表現，但對於處理類別性的變數則較為困難，且應用上對於它學習過程的解讀，比其決策樹演算法較不易直接地了解，造成實際應用上很大限制，所以在解決複雜問題的決

策上,「C4.5決策樹分類學習法」便成為它擅長的領域,因為它自有分岔的條件方法,對離散型或連續型數值中的一個數值可輕易加以分類,所以明顯的優勢即在於處理複雜問題上所產生的決策規則,對於人來說是容易理解的結構[29],且在學習過程中能夠進行調整,及快速被建立,這使得決策樹分類學習法能夠更適用於明確決策模型的相關領域。

下節在我們的方法中,將以步驟說明C4.5分類學習法對連續性數值屬性的處理過程,與如何利用C.45建構使用者行為模型,直至產生最後音樂清單的過程作一闡述。

### 3.1 決策樹之建立

當模型在建置時,倘若我們由時間資料分割為固定長的子序列,例如以全日(24時)來劃分,如此將無任何的時間觀念被包含,導致不論上午或下午所安排推薦的歌曲將會是相同;若改以固定每小時或每分鐘作切割單位,在決策樹分類下的結果,可能造成在不同行為的使用者情況下,系統卻以相同的方式作分割處理,這必定導致某些使用者的喜好資料被錯誤地分類;此外,在切割太小的情況下,也可能造成系統另一種負荷,所以我們視時間為連續性的屬性,與其他欄位一同建立決策樹。

決策樹在建置過程中,需要分析資料集的類別分佈,才能選擇出最適當的分割點與分類屬性。因本研究欲處理對象資料集有「時間標記」與「音樂特徵」具高維度性質的屬性,分類時需考慮到資料量可能十分龐大,且真實資料集中可能有資料稀疏或雜訊等問題,若直接以一般ID3決策樹作分割點搜尋時,資料在分割後可能都只有單一筆數值,然而計算資訊貢獻量又偏向取值最大,造成分類出的規則必定產生過度適配(Over-fitting)的問題,且在擷取有意義的知識上也會大大降低。所以之後Quinlan[31]又提出C4.5分類學習法,延伸解決上述ID3所會出現的問題。

因此,在本研究中會利用C4.5分類學習法作為我們建構喜好模型的基礎,其演算法對於連續數值屬性處理的流程如圖5所示,將透過此演算法進一步分述建置使用者模型的步驟。

#### Step1: 初始化區間

由資料庫中的記錄,我們取得各項屬性,如「時間標記」與「內容特徵」不同維度資料,另依「回饋評價」作為決策樹目標分類的依據。

初期,由於C4.5無法直接對連續型數值資料作處理,所以演算法中會預先把原始的連續

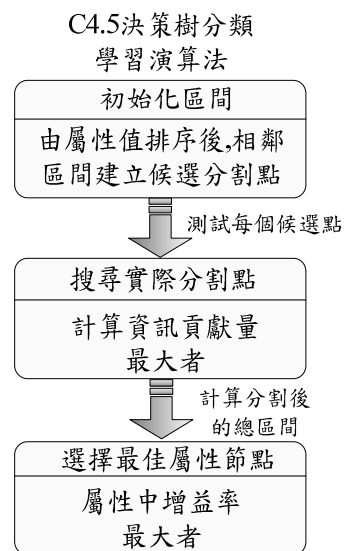


圖5 c4.5於連續數值屬性之處理流程

數值作離散化,再依據其F屬性中的值由小到大排序後,如: $F = \{A_1, A_2 \dots A_n\}$ 。當遇到相同屬性值時,則被視為同一區間的資料集,進一步由各區間之間我們去建立一個候選分割點(Cut Point, CP),此點可由兩相鄰區間計算平均值得到如公式(1)定義:

$$CP = (A_i, A_{i+1}) / 2 \dots \dots \dots (1)$$

上述公式中 $A_i$ 為區間中的連續數值, $A_{i+1}$ 為 $A_i$ 相鄰區間中的連續數值,CP則是分割點的值。

#### Step2: 搜尋實際分割點位置

在F屬性中若找到的區間共有N項,則在各項數值區間中我們建立一個候選分割點,即共取N-1個候選分割點,由資訊貢獻量(Information Gain)最大者來決定實際分割點位置,經此分割方式即能讓資料集達到比分類前更好的分佈。其Information Gain計算方式為「分割前該節點的亂度」減去「分割後子節點的亂度」如公式(2)定義:

$$InfoGain(F) = Entropy_b(S) - Entropy_a(S) \dots \dots (2)$$

若要計算F屬性的資訊貢獻量,則需計算其中分割後的Entropy如公式(3)所示,當S事件中包含N種結果, $C_p$ 則為每種結果所對應到類別的機率,此S事件的亂度為:

$$Entropy_a(S) = \sum_{k=1}^n -C_p \log_2 C_p \dots \dots \dots (3)$$

經由計算各分割點結果並比較其資訊貢獻量的值,將可找出一點K作為實際分割點位置,最後由此點以二元分割方式,把屬性的數值資料分為「小於或等於K」及「大於K」兩部分。



**Step3：選擇最佳特徵屬性節點**

由二元分割後的結果，交由增益率函數計算此屬性的重要性，若 S 事件包含有一屬性 F，當要計算此屬性的 Gain Ratio 值，此計算方式為：「分割前該節點的 Entropy」減去「分割後各子節點的整體 Entropy」再除以「分割後整體的 Entropy 值」，則屬性 F 的 Gain Ratio 如公式(4)所示：

$$GainRatio(F) = \frac{Entropy_b(S) - Entropy_a(S)}{Entropy(A)} = \frac{InfoGain(F)}{Entropy(A)} \dots(4)$$

在選擇最佳「屬性」當決策樹中的節點時，將由不同維度的特徵屬性計算比較 Gain Ratio，選擇最大者優先當樹根(Root)節點。並再由此屬性分類後的子集，以遞迴重覆 Step1~Step3 的演算步驟，在不斷搜尋最佳分割點與 Gain Ratio 最大者的過程，決定樹的下一個屬性節點(Node)與分支(Breaches)，直到分割後每個子集中的都屬於同一種類別或是子集中無資料可再供分類時，則停止分類的動作。

**3.2 使用者行為模型的建置**

由圖6所示為單一使用者的行為模型。在模型產生後，C4.5將會進一步做適當樹的修剪[31]，為避免產生過度學習，會採用事後修剪(Post-Pruning)的方式，作法是由葉節點往上測試所隸屬的子樹節點，並採用預估錯誤率(Predicted Error Rate)為評斷修剪的條件。

此條件定義為某一子樹的訓練資料數量若為N，其中當有K個資料被錯誤分類的機率可表示為K/N，以此來判斷當有新測試資料被決策時不正確的機率值。其作法是給定一個信度(Confidence)，採用二項式分配機率值的上限  $Ucf(K,N)$  作為預估錯誤率的值，其中N代表訓練資料中屬於此子樹的個數，K為N中錯誤分類的資料個數，其預測錯誤率的資料個數則為  $N \times Ucf(K,N)$ ，當子樹節點以葉節點代替後，錯誤預估率結果較低時，則進一步由子樹節點修剪成葉節點，修剪後會以子樹中佔多數的類取而代，反之則保留原來子樹。

由於決策樹建立後，除了希望能具有高的準確性外，最好還能產生簡單及容易解讀的規則(Rule)，所以在修剪決策樹後，將會轉為簡單的程序規則，此規則可從根節點(Root)至各葉節點(Leaf)位置中找到。例如：If (Time) <= 6 : 00 And (Avg\_Pitch) <= 302 And (Time) <= 3 : 25 Than Class = 6 Else Class = 4，並由此依序走訪其它路徑，產生其它的的程序規則，最終轉化後的程序將能夠降低判斷類別時的複雜度。

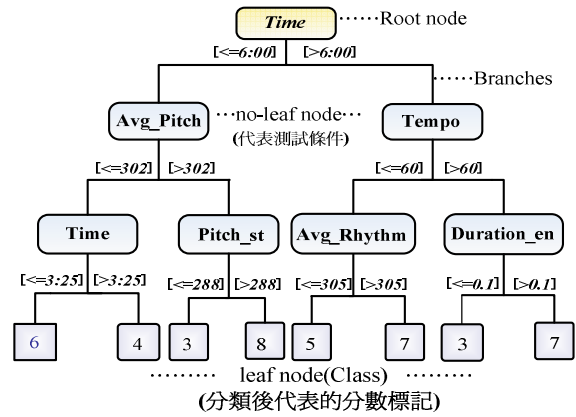


圖6 單一使用者行為模型示意

**3.3 產生音樂排程清單結果**

為避免逐次挑選時每次挑選到同一首歌曲，將由系統至音樂資料庫中隨機選取歌曲，在決策前，經由挑選的音樂依其中內含特徵值資料，比對當前線上使用者的喜好模型，由分類預測結果得到此推薦歌曲可能被喜好的程度，即為歸屬的分數類別，進而由分數高的優先決定我們音樂推薦清單的排序方式，若歌曲在喜好模型中找不到其規則，表示此音樂的內容或當前時段不適合被推薦，這時系統將捨棄它，並繼續挑選其它歌曲做比對動作。

清單產生同時，系統將透過使用者的回饋資料來反應使用者最近在某些時段或聽歌喜好行為的變動情形，以作為下次模型學習或調整推薦內容時的重要依據。

**3.4 改進灰姑娘問題之策略**

此外在推薦過程中，我們發現當使用者在24點後的時段，若之前皆無聽過任何音樂資料情況下，可能受時段與音樂類別的分類結果，使原本應該繼續安排推薦午夜時段中某類型音樂，很可能會受到清晨時段音樂類別的影響，發生一反常態的推薦行為，這裡我們稱它作灰姑娘(Cinderella)問題，以圖7表示。例如原本午夜當前正在聽慢節奏的歌曲類型，若使用者繼續逗留超過午夜12點時，在之前午夜的推薦效力將消失，轉為推薦靠近清晨時段可能是快節奏的音樂類型。這在現實情況下，似乎午

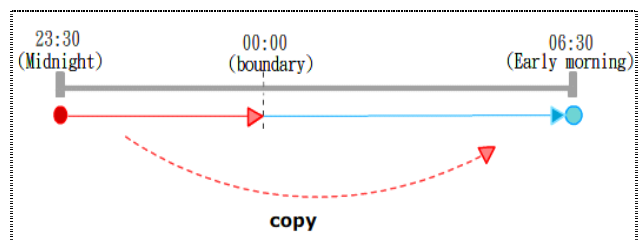


圖7 灰姑娘(Cinderella)問題

表 2 音樂資料庫分類表

編號	風格	數量
1~20	搖滾音樂	20
21~25	爵士音樂	5
26~35	藍調音樂	10
36~50	抒情音樂	15
51~60	古典音樂	10
61~90	國語流行	30
91~115	西洋流行	25

夜時段的音樂才是真正接近目前使用者的喜好，所以為了修正此一問題，當每逢系統跨越午夜24點後，且又在之前訓練的過程中是沒有資料被訓練的情況下，我們將改以複製前午夜時段內容的歌曲。

此時，若我們所設定複製的時段過長，反倒會造成喜好的差距又被拉遠的狀況發生，所以針對此問題進一步改進我們的演算法策略，預設以Random方式，複製前午夜時段(23:30~00:00)中一半的音樂資料，至清晨時段(00:00~06:30)中無資料的時間區段；反之，午夜時段無資料時，則複製清晨時段資料至午夜時段。經由複製的音樂記錄中標注的喜好分數與音樂特徵值，即可在產生使用者決策樹時，解決邊界(24:00)區段無資料的問題。

## 4. 實驗結果

### 4.1 系統環境建立與評估

我們實作以Web環境為主的音樂播放平台，使用者可以利用瀏覽器使用到離線系統所提供的個人服務，經由使用者回饋的資料，以決策樹進行行為模型建立，最後對推薦準確率進行驗證評估，分析其中原因。

本實驗系統的建立採微軟Microsoft Visual Studio 2005環境，以C#為開發程式語言，後端資料庫系統採用Microsoft Office Access 2003，Client端以IE5.0以上瀏覽器，音樂播放器以Macromedia Flash 8.0作為主要人機介面，硬體環境為Pentium 4 1.5G CPU、2G RAM作業系統為Window XP SP2。

歌曲來源是由網路上經有音樂背景者所歸類的MIDI歌曲，並將所蒐集到的歌曲，經由工具程式處理轉為可比對的格式後，存放於音樂資料庫中，MIDI檔歌曲共有115首，其中類別項目包含：搖滾樂/重金屬、爵士、藍調、抒情、古典樂、流行、鄉村、Rap、舞曲、靈

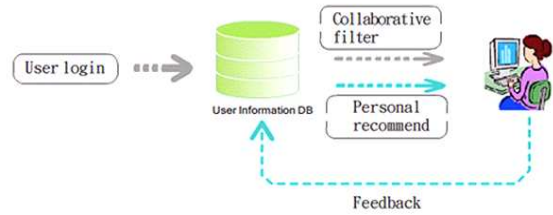


圖8 使用者登入流程

魂樂等...，如表2所示。透過網路與遠端資料庫作連線，當推薦系統建立完成後，除了本身已存在的樣本母體外，之後還會加入新的音樂資料以供歸類，讓使用者能夠在新增的紀錄資料及回饋評價數量累積到一定程度時，自動由代理人調整決策模型作即時決策反應。

### 4.2 系統運作流程

使用者在登入實驗系統後，系統隨即到使用者資訊檔(User Information DB)中利用離線時間找出同好名單，進行合作式過濾推薦，由取得的評價與特徵資料建立並產生行為模型，最後根據模型的特性作個人化的推薦服務如圖8所示。

### 4.3 系統畫面

我們使用Flash建立MuPA(Music Playlist Agent)推薦系統平台，由前端所取得的現況資料，傳送參數至後端核心程式作處理，進一步把結果回傳給Client端的使用者界面。圖9為使用者註冊的畫面，使用者可透過此介面進行註冊，爾後使用者可以根據聽過相關的推薦音樂，給予不同實際回饋，如圖10所顯示的播放清單與評比介面。最後，產生使用者的喜好模型後，在Menu中提供適性的個人化的推薦清單。



圖9 使用者註冊畫面



圖10 MuPA推薦清單與評比介面

#### 4.4 實驗結果與分析

從學習的資料庫中我們作一分析，得到以下樣本資料的統計結果，參與本實驗的註冊人數55人，本系統中音樂總樣本數達初步規模，基本資料統計得知男女的比例為8：2，年齡分佈在23歲到45歲之間，為網路高度使用族群，目前持續實驗希望由不斷增加的學習資料來更進一步分析其中結果。

一般若要實驗系統的回應率時，必須是在使用者聽完資料庫中的所有樣本資料的前提下，才能對此項目進行實驗，但礙於線上使用者使用情況的不同，我們無法每個人都要求做到，在此回應率的實驗部份我們就不多加探討，所以為了驗證本研究提出以「時間排程」為概念的系統架構，將透過實證方式得知其中結果。評估系統的準則我們將採用平均絕對誤差(MAE)與準確率(Precision)兩種指標，此兩種指標已被廣為用在各類推薦系統的評估標準上Mobasher[3]、Herlocker[14]、Sarwar[37]、Vucetic [40]、Yu[44]，以下我們將利用上述評估指標分別對各個實驗作詳細說明，並在最後列出我們的實驗結果與分析討論。

- 實驗一、建構決策模型系統亦會評估該訓練資料集的錯誤率。

錯誤率(Error Rate)指的是經由各種分類技術分析並建構模型後，模型對於原訓練資料錯誤分類之比例，我們將透過此MAE指標來檢視，其主要目的為評估預測的評比與實際評比間平均的誤差值，其MAE的公式定義如下。

$$MAE = \frac{\sum_{u=i}^N |P_{u,i} - R_{u,i}|}{N} \dots\dots\dots(5)$$

其中N為測試的資料筆數，而 $P_{u,i}$ 為使用者u

表3 三回合平均MAE的試算

User-id	第一回合		第二回合		第三回合		
	P-R	R-R	P-R	R-R	P-R	R-R	
21	1	3	2	3	2	3	
21	2	3	2	3	2	2	
21	3	2	3	5	2	4	
21	3	2	4	5	2	2	
21	7	8	8	8	6	7	
MAE(%)		1.2		1		0.8	
Avg-MAE (%)						2.46	

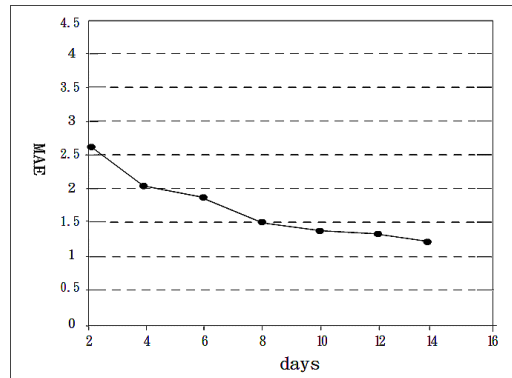


圖11 時間排程推薦之MAE值

對音樂項目i的預測評分； $R_{u,i}$ 為使用者u對音樂項目i的實際評分。

實驗作法將隨機挑取使用者的資料對本系統作測試，讓每位使用者在推薦前要求聽15首以上的歌曲，方便建立個人的行為資料檔，其中選取3/5筆音樂資料為我們訓練資料，另外其它音樂資料當測試，兩者為不重複的音樂資料集，以作為前測之判斷，我們將透過範例表3計算說明。

表中「P-R」、「R-R」分別代表預測類別與實際類別，顯示前5筆系統推薦的音樂歌曲項目，表中分別表示得到的系統預測與實際評比結果，由MAE公式計算得之第一回合結果： $(1-3)+(2-3)+(3-2)+(3-2)+(7-8)/5=1.2$ 。以此類推分別計算二、三回合，最後可得三回合平均絕對誤差為2.46，藉此計算方式我們得到本實驗結果，如圖11所示。

由圖中週期內MAE值結果看出，初期MAE的值呈現下降趨勢，一開始的表現誤差較大，隨時間的拉長，同樣地使用回饋資料也不斷增加下，可以發現MAE的值開始有較大幅度的下降，到後來趨於平緩的表現，此實驗系統約在8天後MAE的值能達一個穩定的狀態。

- 實驗二、我們針對Yoshinori Hijikata[43]提出以C4.5決策樹建構下的推薦系統、Random推薦方法與本研究的方法比較其推薦準確率。

表4 三回合平均準確率的試算

User-id	第一回合		第二回合		第三回合	
	R-R	convert	R-R	convert	R-R	convert
21	2	0	1	0	2	0
21	6	1	3	0	3	0
21	6	1	6	1	6	1
21	8	1	7	1	7	1
21	8	1	7	1	8	1
Precision (%)	80		60		60	
Avg-precision (%)	66.66					

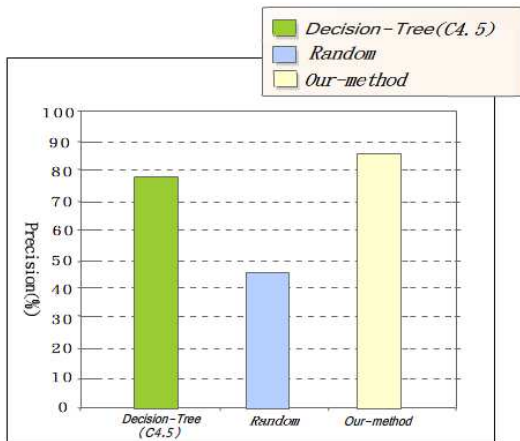


圖12 各分類演算法準確率之比較

推薦準確率(Precision)指的由各種的分類演算法分析，在建完模型後，此模型對於測試資料能準確推薦之比例。其準確率的公式定義如下：

$$Precision = \frac{\text{Number of accept items}}{\text{Number of recommend items}} \dots\dots(6)$$

準確率公式中，主要是以系統推薦出的音樂且被使用者所接受的音樂數量(Number of accept items)，相對於所有推薦數量(Number of recommend items)，也就是系統所推薦的音樂項目當中，有多少的比例是使用者當前有興趣的。

我們驗證的作法，先由系統自動推薦給目前使用者適合的音樂，以前15筆推薦出的資料為評估重點，為達客觀性將重覆進行三回合評估計算，並透過此規則「訓練學習-模型建立-準確率計算」來進行平均準確率的計算。在本研究系統中最大評比為十分，若回饋資料是一到五分的結果，我們將標記為「0」即為「不喜歡」；若為六到十分的結果則標記是「1」即為「喜歡」，所以當計算得到Precision的值越高代表系統所推薦的音樂項目越受使用者認同，我們將透過範例表4計算說明。

範例中顯示前5筆系統推薦出的音樂歌曲項目，其中第一回合的回饋的評比一到五分占

其中總筆數4筆，即標記為「0」表示不喜歡；反之，標記為「1」表示喜歡，此表中為受測者ID(21)的使用後情形，計算三回合得知其平均推薦的準確率結果為： $(4/5*100)+(3/5*100)+(3/5*100)/3=66.66\%$ 。最後，我們由此方式得到如圖12所示的實驗結果。

由長條統計圖可看出於C4.5決策樹與我們加入時間排程概念下的方法平均都在75%以上的準確率，比較下我們的方法推薦準確度較好，相對隨機推薦方式的滿意度表現平均準確率低於50%，從這裡得知若以random方式得到結果較不受使用者滿意，可見所提出的「時間排程」系統架構，是有助於個人推薦能力的提升。

- 實驗三、比較不同類型使用者的使用滿意度。我們分別針對學生、教師、上班族、家庭用戶比較推薦準確率的差異。

驗證方式大致與實驗二相同，主要目的在針對不同類型使用者的使用狀況，分析因為聽音樂的地點或時間的差異情形下，在我們所提出的時間排程推薦系統架構中，是否能夠更有效將音樂適時推薦給符合的使用者，我們將依四種類型的使用者，隨機挑選各四名不同類型用戶作為受測者，在試算表5中，我們以「Student」類型用戶為計算範例說明。由範例可得此受測者ID(6)三回合的平均準確率為： $\{(3/5*100)+(2/5*100)+(3/5*100)\}/3=53.33\%$ ，由此方式驗證得到我們實驗結果，如圖13所示。

表5 三回合平均準確率試算表

User-id (6)	第一回合		第二回合		第三回合	
	R-R	convert	R-R	convert	R-R	convert
Student	2	0	1	0	2	0
Student	2	0	3	0	3	0
Student	6	1	4	0	6	1
Student	7	1	6	1	6	1
Student	8	1	7	1	7	1
Precision (%)	60		40		60	
Avg-precision (%)	53.33					

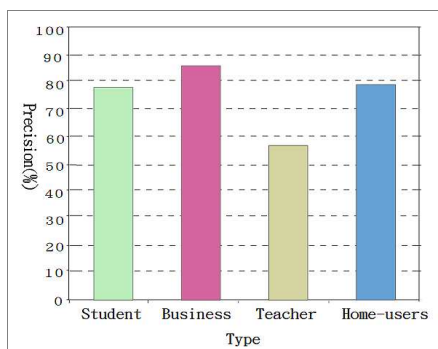


圖13 不同類型使用者於準確率的表現

經由不同類型用戶使用後準確率的表現，從圖中可看到大致上本研究的音樂推薦方式，在不同類型用戶的平均準確率都有在80%以上，亦即非常滿意。尤其「上班族」的準確率表現最佳，原因可能因為受測者的上班/下班的時間較為規律，且在某些時間安排的音樂都較接近受測者喜好所致，此結果達成原先預定中上的滿意程度，表示本音樂推薦系統融入時間排程的概念，對於適時適地的音樂安排是可行的作法，且準確率方面也有不錯的成效。

#### 4.4 研究限制

1. 本研究所收集的音樂樣本資料為115筆，在統計上樣本的資料收集仍稍嫌不足，往後若有更充裕的樣本資料，在未來作研究上將更具完整性。
2. 實驗中，礙於時間關係無法作到長期觀測與學習的目標，在往後將可拉長時間作更深入的分析及研究。

#### 5. 結論與建議

本研究由推薦系統的相關理論與其中方法的探討，提出以時間排程為考量的創新推薦方式。總言之，本研究主要特色與貢獻如下：

- (1) 解決Cold-start問題：以往音樂推薦系統在新使用者的推薦上也常遇到相同問題，一旦系統在無任何評價與資訊不足情況下，往往開始時不能達到適合的推薦效果。所以本研究以啟始註冊的新使用者所提供的相關喜好資訊作為社群分析以找到同好的使用者，來達到初始無法有效推薦的缺點。
- (2) 加入時間概念的推薦方法：在一般推薦系統代理人的作法，脫離不了以使用者瀏覽行為作推測。然而上述所產生的清單推薦，系統最多只能滿足使用者的相似內容項目的取得。則本研究以混合式推薦為基礎，再加入時間的新概念，透過決策樹不斷地訓練及學習過程，將可改善無法提供使用者因喜好隨時間變動而無法自動調適的缺憾，由此系統將能帶給使用者更貼近他們生活的音樂資訊，使個人化的音樂推薦服務能更具完備。
- (3) 解決灰姑娘問題：由於在音樂播放清單中注入時間的概念，在實驗過程中，發現當午夜時段在跨越清晨時段的時候，若有一邊時段發生無訓練資料時，則系統會以較靠近清晨時段的歌曲類型為推薦依據，針對此異常的推薦行為，我們於本研究中提

出一個複製的策略，加以改進解決決策樹演算法中的缺失，並能在往後發生於同一時間區段時，能作出合理的決策。

#### 5.1 未來研究與建議

由實證分析結果，證實本研究所提出的個人化音樂排程代理人已有初步成效，未來在個人化推薦研究的路上，仍有以下可待改善或供後續研究發展的方向：

1. 推薦演算法的改良：本研究主要以決策樹為核心架構，之後可改以其他的學習演算法，如類神經網路、基因演算法、或k-means等探勘技術，更進一步提升或改善其預測結果。
2. 系統功能改善方面：目前特徵擷取步驟皆採人工方式建置，若能發展一套自動化內容特徵擷取模組，將能提高系統功能的時效性及完整性。
3. 概念擴展：由於音樂治療(Music Therapy)在國外實施已行之有年，其透過聆聽的過程，可幫助治療患者達到解除緊張情緒，減輕患者痛苦，進而增進健康復原的處方，所以未來也可嘗試把此離型系統的應用移轉到醫療院所，設計一套專屬病患的音樂排程系統。

#### 參考文獻

- [1] Amazon.com, "Earth's biggest selection, includes personalized recommendation and wish lists for registered users," <http://www.amazon.com>.
- [2] Arthur Flexer, Dominik Schnitzer, Martin Gasser, Gerhard Widmer, "Playlist Generation Using Start and End Songs", (ISMIR) 2008.
- [3] Beth Logan., "Music recommendation from song sets.", (ISMIR) ,pp. 425-428, 2004.
- [4] Bamshad Mobasher1, "Honghua Dai, Tao Luo, Yuqing Sun, Jiang Zhu, "Combining Web Usage and Content Mining for More Effective Personalization." In *Proceedings of the International Conference on E-Commerce and Web Technologies*, 2000.
- [5] Bello, JP, L. Daudet and MB Sandler, "Time-domain Polyphonic Transcription using Self-generating Databases," in *Proceeding of the 112th convention of the audio engineering Society*, 2002.
- [6] Claudiu S. Firan, Wolfgang Nejdl, Raluca Paiu "The Benefit of Using Tag-Based Profiles.", 2006.

- [7] Daniel McEnnis, Sally Jo Cunningham, "Sociology And Music Recommendation Systems", 2006.
- [8] Elias Pampalk, Tim Pohle, Gerhard Widmer "Dynamic Playlist Generation Based on Skipping Behavior", (ISMIR) 2005.
- [9] Fu, Y. "Data mining: tasks, techniques, and applications." *IEEE Potentials*, Vol. 16, No. 4, pp. 18-20, 1997.
- [10] Fayyad, U.M. & Irani, K.B. "On the handling of continuous-valued attributes in decision tree generation." *Machine Learning*, Kluwer Academic Publishers, Massachusetts, pp. 87-102, 1992.
- [11] Goldberg, D., D. Nichols, B. M. Oki, and D. Terry, "Using Collaborative Filtering to Weave an Information Tapestry," *Communications of the ACM*, 1992.
- [12] Hung-Chen Chen, Arbee L.P. Chen, "A music recommendation system based on music data grouping and user interests," *Proceedings of ACM International Conference on Information and Knowledge Management*, 2001.
- [13] Herlocker, J. L. & Konstan, J.A., "Content-Independent Task Focused Recommendation," *IEEE Internet Computing*, pp. 40-47, 2001.
- [14] Hammer, Eleanor Ray & COLE, Malcolm S. Cole "Guided listening: a textbook for music appreciation.", 1992.
- [15] Herlocker, J., Konstan, J. A., Borchers, A., & Riedl, J. "An Algorithmic Framework for Performing Collaborative Filtering", *Proceedings of SIGIR 99*, Berkeley, pp. 230-237, 1999
- [16] *ilike.com*, "http://www.ilike.com," Retrieved date: 2007/11/5 (2007)
- [17] K. Lang, "Newsweeder : Learning to Filter Netnews," *Proceedings of the 12th International Conference on Machine Learning*, pp: 331-339, July, 1995.
- [18] Konstan, J., Miller, B., Maltz, D., Herlocker, J., Gordon, L., and Riedl, J., "GroupLens: applying collaborative filtering to use net news.," *Communications of the ACM* 1997.
- [19] Katsuhiko Kaji, Keiji Hirata, Katashi Nagao, "A Music Recommendation System Based on Annotations about Listeners," 2005.
- [20] Kuo, F.-F. and Shan, M.-K. "A Personalized Music Filtering System Based on Melody Style Classification," *IEEE International Conference on Multimedia and Expo ICME*, pp. 9-12, Dec. 2002.
- [21] Lu, L., Jiang H. and Zhang, H., "A robust audio classification and segmentation method," *ACM Multimedia* pp. 203-211, 2001.
- [22] *last.fm*, "http://www.last.fm," Retrieved date: 2007/11/6 (2007)
- [23] M. Balabanovic & Y. Shoham, "Learning Information Retrieval Agents: Experiments with Automated Web Browsing," *AAAI Spring Symposium on Information Gathering from Heterogeneous, Distributed Environments*, Stanford, 1995.
- [24] McKinney, M. F., and Breebaart J, "Features for Audio and Music Classification," (ISMIR) 2003.
- [25] Monti, G. and M. Samder, "Pitch Locking Monophonic Music Analysis," in *Proceeding of the 112th convention of the audio engineering Society*, 2002.
- [26] M Slaney, K Weinberger, W. White, "Learning a Metric for Music Similarity" (ISMIR) 2008.
- [27] Nuria Oliver, Lucas Kreger-Stickles "PAPA: Physiology and Purpose-Aware Automatic Playlist Generation", (ISMIR) 2008.
- [28] Oscar Celma, Miquel Ram´rez, Perfecto Herrera, "A MUSIC RECOMMENDATION SYSTEM BASED ON RSS FEEDS AND USER PREFERENCES," 2005.
- [29] Po-Chuang Huang, "Finding Multi-interval Classification Rules in Continuous valued Attributes : A Decision Tree Approach ", *Master's dissertation of National Cheng Kung University, (in Taiwan)*, 2004.
- [30] *Pandora.com*, "http://www.pandora.com," Retrieved date: 2007/10/20 (2007)
- [31] Quinlan, J. R. "C4.5: Programs for Machine Learning", *Morgan-Kaufmann, San Francisco*, 1993.
- [32] Raphael, C., "Automatic Transcription of Piano Music," in *Proceedings of the 3rd International Symposium on Music Information Retrieval (ISMIR'00)*, 2000.
- [33] Rob van Gulik & Fabio Vignoli, "Visual Playlist Generation on the Artist Map" *Institute of Information and Computing*

- Sciences, Utrecht University, (ISMIR) 2005.*
- [34] Shan, M.-K. Kuo, F.-F. and Chen, M.-F. "Music Style Mining and Classification by Melody," *IEEE International Conference on Multimedia and Expo ICME*, Vol. 1, Aug, pp.26-29, 2002.
- [35] Steffen Pauws and Sander van deWijdeven, "User evaluation of a new interactive playlist generation concept". *International Conference on Music Information Retrieval, ISMIR*, pp: 638-643, 2005.
- [36] Shardanand, U., Maes, P., Social information filtering: algorithms for automating "word of mouth." *In Proceedings of the ACM CHI Conference*, 1995.
- [37] Sarwar, B. M., Karypis, G., Konstan, J. A., & Riedl, J. "Application of Dimensionality Reduction in Recommender System-A Case Study", *ACM WebKDD Web Mining for E-Commerce Workshop*, 2000.
- [38] Srinivasan, S., Petkovic, D. and Ponceleon, D. B., "Towards robust features for classifying audio in the CueVideo system," *ACM Multimedia*, pp:393-400, 1999.
- [39] The Echo Nest Analyze "API.<http://developer.echonest.com/docs/analyze/xml>" xml description, downloaded March 31, 2008.
- [40] Vucetic, S. & Obradovic, Z. "A regression-based approach for scaling-up personalized recommender systems in e-commerce", *ACM WebKDD Web Mining for E-Commerce Workshop*, 2000.
- [41] Wang, H., Wang, J., Yang, J., and P. S. Yu, "Clustering by Pattern Similarity in Large Data Sets", *Proceeding of ACM Special Interest Group on Management of Data*, 2002.
- [42] Y.S. Tzeng, H.C. Chen, and A.L.P. Chen, "A New Approach for Rating-Based Collaborative Music Recommendation Using Personal Preferences and Opinions From Trustable Users," *Proc. of The IASTED Conference on Internet and Multimedia Systems and Applications (IMSA)*, 2004.
- [43] Yoshinori Hijikata, Kazuhiro wahama, Kazuki Takegawa, Shogo Nishida, "Content-based Music Filtering System," *(ACM)*, 2006.
- [44] Yu, K., Xu, X., Ester, M., & Kriegl, H.P. "Selecti

ng relevant instances for efficient accurate collaborative filtering". *Proceedings of the 10th international conference on Information and knowledge management*, pp:239-246, 2001.