

# 行政院國家科學委員會專題研究計畫 期中進度報告

## 數位家庭助理之研究與實作(1/3)

計畫類別：個別型計畫

計畫編號：NSC94-2213-E-004-010-

執行期間：94年08月01日至95年07月31日

執行單位：國立政治大學資訊科學系

計畫主持人：陳良弼

報告類型：精簡報告

處理方式：本計畫可公開查詢

中 華 民 國 95 年 6 月 1 日

行政院國家科學委員會補助專題研究計畫  成果報告  
 期中進度報告

### 數位家庭助理之研究與實作(1/3)

Research and Implementation of Digital Home Assistants

計畫類別： 個別型計畫  整合型計畫  
計畫編號：NSC-94-2213-E-004-010  
執行期間：94年08月01日至97年07月31日

計畫主持人：陳良弼

成果報告類型(依經費核定清單規定繳交)： 精簡報告  完整報告

本成果報告包括以下應繳交之附件：

- 赴國外出差或研習心得報告一份
- 赴大陸地區出差或研習心得報告一份
- 出席國際學術會議心得報告及發表之論文各一份
- 國際合作研究計畫國外研究報告書一份

處理方式：除產學合作研究計畫、提升產業技術及人才培育研究計畫、列管計畫及下列情形者外，得立即公開查詢

涉及專利或其他智慧財產權， 一年  二年後可公開查詢

執行單位：國立政治大學資訊科學系

中華民國 95 年 05 月 31 日

# 數位家庭助理之研究與實作(1/3)

## Research and Implementation of Digital Home Assistants

計畫編號：NSC-94-2213-E-004-010

執行期間：94年08月01日至97年07月31日

計畫主持人：陳良弼

執行單位：國立政治大學資訊科學系

### 中文摘要

目前許多的多媒體資料很容易透過網路取得。使得過去僅存在於公司才有的資訊過量問題，也漸漸在一般的日常生活中產生。因此，如何幫助一般的使用者在眾多的多媒體資料中，快速的找出其心目中想要的資料，來達到優化生活的目標，變成一個重要的課題。本計劃考慮了三大類的多媒體資料，即：音樂、影片和時序性資料。依其特殊需求研擬可行的查詢或探勘方法，而最後，我們會將這些方法建置成一個可以幫助一般使用者提升生活品質的系統。

**關鍵詞：**多媒體、資料探勘、查詢、索引、特徵擷取

### 英文摘要

In recent years, multimedia data are easily obtained from the network. Therefore, we can use these data to improve living quality. For example, when a user reads a novel, the family multimedia retrieval system can find a piece of suitable music to raise the reading atmosphere. Therefore, helping the users to efficiently find the wanted multimedia data becomes one of the important research issues. We consider three types of data, i.e., music, video, and time sequencing data, and design querying and mining methods according to their characteristics. Finally, we will integrate the results to build a system to raise the quality of life.

**Keyword:** Multimedia, Data Mining, Querying, Index, Feature Extraction

### 一、前言

近幾年來，隨著網路的發展，許多資訊與多媒體資料都很容易取得。因此，如何幫助使用者分析與整合過多的資訊與多媒體資料，成為一個重要的課題。加上電腦科技的蓬勃發展以及多媒體應用與資料分析的技術逐漸成熟，越來越多的研究人員以及企業致力於數位家庭環境以及相關應用的研究與開發。因此，整合使用者周遭各式不同種類之多媒體資料，並輔以個人化的呈現是數位家庭環境下的一個重要應用。而本計畫的主旨，即在於研究多媒體的查詢與跨媒體整合技術，以及時序性資料的分析與預測，進而提供使用者在數位家庭環境下的多媒體娛樂以及理財資訊支援。

過去的一年裡，在個人化的多媒體資訊整合方面，我們將音樂查詢與播放技術和人類的情感做結合，使得所查詢所得的結果能夠符合使用者的情感需求。而在影像方面，我們也發展了處理影片物件的技術，這可以與音樂方面的技術做整合，為第二年的跨媒體整合做準備。最後，在數位理財方面，我們也設計了一套能夠預測時序性資料的方法，來輔助使用者做決策的判斷。

### 二、研究目的

以下茲就音樂、影像與時序性資料等三大主題說明本計畫於今年度之研究目的：

#### 音樂查詢與播放技術之研究

音樂資料每天都在成長，如何快速地从龐大的音樂中找到自己想要的音樂是個重要的議題。而把音樂資料輔以索引結構是加速查詢的關鍵。但是過多的音樂資

料且音樂資料過長的特性將使得索引結構所需要的空間太大。因此本研究的目的是在於找出音樂的重複片段，並依此將歌曲切成許多小段的樂句，以減少索引結構的大小，而能加速查詢。此外，在索引結構建立的同時，我們還需考慮到歌曲所隱含的情緒特徵，使得查詢的結果能符合使用者的情緒。

### 影像查詢技術之研究

在多媒體資料中，影片資料佔了很大的比重，更重要影片資料是家庭生活中不可缺乏的部份。因此，如何對影片資料做有效的處理變成一個課題。在本研究的目的是提出一套索引方式與演算法，使得系統能依據影片中的物件來做查詢，而且當該物件具有多屬性(例如：影片中的車子具有方向，速度等不同的屬性)時，亦能有效的查詢。以作為第二年跨媒體整合的基礎。

### 時序性資料走勢之分析及預測

在今年度的計畫中，我們藉由分析時序性資料，用以推導出具有高信賴度的關聯法則，以期利用導出之法則來預測時序性資料未來的走勢。本年度計畫主要著重在時序性資料的線上 (online) 分析，我們希望所設計出的系統能夠立即掌握時序性資料的變化，並且在極短的時間內建議使用者勝率較高的操作方式 (買進或者賣出)。而時序性資料的分析結果，可用於輔助家庭的金融理財活動，如：股票投資等。

### 三、文獻探討

在音樂資料方面，[15]針對音樂查詢系統提出三個階段的架構，第一階段是主旋律擷取。由於一般複音的 MIDI 格式音樂歌曲，主旋律無法直接獲得，所以此階段中，最好的主旋律擷取方法為在同時發音的音符中，取音高最高的音符作為主旋律。第二階段標準化則是希望統一各歌曲以及查詢的表示法，以絕對音高及相對音程差的效果為佳。第三階段就是決定查詢

與歌曲的相似度，以 local alignment 為最好。

在音樂分段方面，[1]提出一套 rule based model，根據 Change Rule 找出連續兩音符的改變程度，Proximity Rule 可以用來決定邊界落在 interval 較大的音符上。

在找出音樂重複片段方面，過去有不少研究在找出完全相同的重複樣式 (repeating pattern)[7][13]，但是這用在音樂資料上是不太合宜的。[12]提出一個找出近似重複樣式 (approximate repeating pattern) 的方法，用 sliding window 方式將音樂資料切成許多 segments，每個 segment 跟其他 segments 以 edit distance 為相似度測量標準進行相似度比較，相似的 segments 數目若超過門檻值，便是一個近似重複樣式。[10]延續此一架構，設計了一個 Lower-bounding distance，先使用計算複雜度較低的距離計算方式去過濾不相似的 segments，最後再真正用 edit distance 去計算相似度，用以降低計算時間。

近年來有關音樂與情緒的研究主要可分為兩類：(一)利用音樂與心理方面所建立的模型進行音樂的分類。[2][8](二)根據預先分類好的歌詞資料利用 decision tree 等方式建立歌詞分類法則。[16](三)根據預先分類好的音樂資料利用 SVM 等方式建立音樂分類法則。[9]

而這些方法主要都是在探討所謂音樂本身的情緒，然而，每個人對音樂的感覺會因為種族、成長背景等各項因素而有所不同，甚至於同一個人在不同的時間也會有不同的感受，因此，一個新的音樂情緒系統應該要能夠根據使用者不同的感受而自動找出最適當的情緒音樂，更進一步要能夠找出適合某種情境的感覺音樂。

對於時序性資料的分析，在過去相當多的文獻探討中，片段 (episode) 常被用以表示一事件 (event) 集合間的關係。給予一個時間框架 (time window)，序列片段 (serial episode)  $\langle e_1 e_2 \dots e_n \rangle$ ，用以表示事件  $e_1, \dots, e_n$  其發生的時間順序，其中， $e_n$  的發生時間減去  $e_1$  的發生時間是小於框

架尺寸 (window size)。而平行片段 (parallel episode) ( $e_1, \dots, e_m$ ) 描述在時間框架內事件  $e_1, \dots, e_m$  都有發生，但是不管事件間的發生順序。而合成片段 (composition episode) 為多組序列片段和平行片段的組合，其所有描述的事件也都必須發生在時間框架尺寸內。

在本年度計劃中，我們將一股票的多種技術指標形成多維的事件序列 (event sequence)，我們進而參考過去文獻對頻繁片段探勘 (frequent episode discovery) 的探討，發展出多個能夠線上 (on-line) 推導片段法則 (episode rules) 的演算法，所推導出的法則將具有高信賴度 (confidence)，並且將被應用於線上的股票價格之預測。以下為相關文獻探討：

在分析片段及片段法則方面，在[6]中有兩種計算片段出現次數的方式，一個是利用固定時間長度的滑動視窗，另一個則是在一個固定時間限制內，計算出現該片段的最小區間，然而其子區間沒有出現片段。[6]並且也對兩種計算方式，設計了 WINEPI 和 MINEPI 兩種求取頻繁平行及序列片段的方法。由於合成片段十分複雜，[6]僅設計針對平行與序列片段的探勘。

有了求出頻繁片段的方法之後，若要求出如下片段法則： $A \rightarrow B$ ，前項片段 (antecedent) A 在時間框架 T 發生之後，後項片段 (consequent) B 則有超過 70% 的機率會在時間框架 S 內發生。則必須對 T 與 S 兩個時間限制分別對資料集做挖掘頻繁片段的工作。

上述的找尋片段法則的方式，如果想發現片段 B 有較長的後項延遲時間 (lag time，指前項片段開始時間與後項開始時間的時間差)，這個將會造成計算量大幅度的增加。

對於一片段法則，[5] 將前項片段與後項片段分開，在求出頻繁片段之後，再兩兩尋找其符合信賴度的關聯法則及後項的延遲時間。值得一提的是，本篇與前篇所探討問題之不同處在於前篇只考慮一維的事件序列，而本篇則考慮在同一個

時間有多個維度的事件值。傳統求取片段法則的方法如[6]，如果需要能求出後項較長的延遲，可能會使計算量與延遲長度呈指數關係。在[5]雖然將前項片段與後項片段分開，但是並無提出適合的作法。

另一個相似的環境在[4]被提到，[4]將 MINEPI 的廣度搜尋改成深度搜尋成為 MINEPI+，並且利用記憶體的索引結構而設計了 EMMA 方法來加速探勘頻繁片段的時間。

#### 四、研究方法

以下茲就音樂、影像與時序性資料等主題說明本計畫於今年度所提出之研究方法：

##### 音樂查詢與播放技術之研究

這部份的工作主要又可以細分為兩大項，一是音樂索引的建立和相似樂曲的查詢，二是依情緒／情境的樂曲擷取。這兩個部份是為了第二年的跨媒體整合做準備。以下我們分別介紹。

##### (一) 音樂索引的建立和相似樂曲的查詢

主要可以分成下面的步驟：

##### 步驟一：音樂特徵擷取

1. 主旋律擷取：採取[15]的方法來作主旋律擷取。在音樂資料中，若有同時發音的音符，保留音高最高者為主旋律音符。

2. 設計樂句分割法則及演算法：大部分的樂句都會結束在終止音，而終止音有兩種，第一種終止音在休止符的前一個音符，所以我們先計算每一個音符的開始發音的時間加上它的持續時間，然後去檢查跟下一個音符開始發音的時間中間是否有空檔的時間，若有空檔時間且時間超過一個門檻值，這裡就當做有一個休止符，所以先用與有找到的休止符將音樂資料作第一步的分割。第二種終止音是音符的持續時間突然變得較長，我們用[15]的演算法套入每個音符的持續時間找出這種終止音，此為第二步分割。最後分割完成的每個段落便視為一個樂句。

3. 發展自動化擷取音樂特徵之方法：我們以一個新的方式去自動找出音樂

資料當中的近似重複片段。利用[3]當中提到一個以edit distance為基礎的近似字串比對方法(公式一)，這個方法可以找出被比對字串(T)中，局部與查詢字串(P)最相似的部分，假設P是 1234，T是 123423123143，利用公式一可以得到表格一，在最後一列(第四列，有顏色部分是產生的表格內容)的第四行可以知道在T中，有一片段與P相似，其edit distance為0，並且可以知道這片段結束在T的第四個位置( $t_4$ )。另外可以在利用公式二同時產生的表格二中，相對於表格一的位置(第四列第四行)得到一個值為4，這代表在T中與P相似的這個片段長度為4，又我們知道它結束在 $t_4$ ，所以可推得此在T中與P相似的片段為 $t_1 t_2 t_3 t_4$ 。

$$\begin{aligned}
 &P = p_1 p_2 \dots p_m, T = t_1 t_2 \dots t_n \\
 &D[0, j] = 0, \quad 0 \leq j \leq n \\
 &D[i, 0] = i, \quad 0 \leq i \leq m \\
 &D[i, j] = \min \begin{cases} D[i-1, j] + 1 \\ D[i-1, j-1] + 1 \text{ (if } p_i = t_j \text{ then 0 else 1)} \\ D[i, j-1] + 1 \end{cases} \\
 &\text{for } 1 \leq i \leq m, 1 \leq j \leq n
 \end{aligned}$$

公式一

$$\begin{aligned}
 &L[0, j] = 0, \quad 0 \leq j \leq n \\
 &L[i, 0] = 0, \quad 0 \leq i \leq m \\
 &L[i, j] = \begin{cases} \text{if } (D[i, j] = D[i-1, j] + 1) \text{ then } L[i-1, j] \\ \text{elseif } (D[i, j] = D[i-1, j-1] + 1 \text{ (if } p_i = t_j \text{ then 0 else 1)}) \text{ then } L[i-1, j-1] + 1 \\ \text{else } L[i, j-1] + 1 \end{cases} \\
 &\text{for } 1 \leq i \leq m, 1 \leq j \leq n
 \end{aligned}$$

公式二

我們的方法就是把音樂資料當 T，然後也是用 sliding windows 的方式從音樂資料切出片段來當 P。延續前面的例子，假如音樂資料 T 為 123423123143，sliding windows 大小為 4，第一個 P 就是 1234。然後可以看表格一的第四列中，第四行的 0 表示這裡有一個片段與 P 完全相似。第九到十一行的 1 表示這附近有三個重疊很大的片段與 P 相似，且與 P 距離為 1。這種情形，我們先找距離最小的片段當代表，但在此例中，最小的片段有三個，就去表格二相對的位置，找出長度最長的片段當代表，所以這個在 T 中與 P 距離 1 的

片段代表就是 12314。

		1	2	3	4	2	3	1	2	3	1	4	3	1
	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	1	0	1	1	1	1	1	0	1	1	0	1	1	0
2	2	1	0	1	2	1	2	1	0	1	1	1	2	1
3	3	2	1	0	1	2	1	2	1	0	1	2	1	2
4	4	3	2	1	0	1	2	2	2	1	1	1	2	2

表格一

	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	0	1	0	0	0	0	0	1	0	0	1	0	0	1
	0	1	2	3	0	1	0	1	2	3	1	2	0	1
	0	1	2	3	4	1	2	1	2	3	4	2	3	1
	0	1	2	3	4	5	2	3	2	3	4	5	3	4

表格二

在[10][12]都可以去定義感興趣的近似重複樣式大小，也就是 sliding window 的大小，我們的方法同樣可以處理，而且不會多花太多的時間。例如前面例子，若使用者感興趣的近似重複樣式大小為 2 到 4，我們直接以 4 當作 sliding window 的大小，而 2 跟 3 的結果一樣可以從同一個表格的第二列跟第三列得到。

4. 原型旋律之擷取及驗證：在上一步中找到的近似重複樣式，會有數個與它近似的片段，我們以按順序且同時出現在所有片段的音符為我們的原型旋律。

### 步驟二：索引結構設計

1. 定義適當之音樂表示法：以相對音程差(後面音符的音高減前面音符的音高)來當作音樂的表示法。

2. 主旋律索引建立：將所有音樂資料的主旋律轉成定義的音樂資料表示法，然後建在同一個 suffix tree 中。

3. 樂句式 (phrase-based) 索引研究：將所有音樂資料的主旋律先做樂句分割，再轉成定義的音樂資料表示法，然後建在同一個 suffix tree 中。

### 步驟三：近似查詢處理

1. 相似度計算函數訂定：以 edit distance 為相似度測量標準。

2. 查詢演算法之設計：自動產生所有與查詢字串的 edit distance 在門檻值內的字串，然後一一去 suffix tree 中做完全比對(exact matching)。

### (二) 依情緒／情境的樂曲擷取

以情緒／情境為基礎的音樂檢索與過去音樂檢索最大的不同在於必須建

立一個新的音樂表示法，不同於過去將音樂資料建立成索引後，利用一段旋律等音樂片段來進行相似音樂搜尋的方法，在檢索情緒／情境時，我們必須要利用各種特徵來表示音樂的特性，目的在於找出什麼樣特性的音樂會有什麼樣的情緒／情境效果。主要方法可分為以下幾個步驟：

#### 步驟一：建立新的音樂表示法

過去在表示一首音樂的特徵時，最常使用的方法就是平均，例如我們可以說這首音樂的平均音高是多少？平均音長是多少？平均音量是多少？藉此來表示一首音樂的特性，希望能藉此找到一些分類規則，例如音量在哪些範圍時分別代表哪些情緒／情境等，然而，平均卻也有著很多缺點：

- (1) 容易被一些較大或較小的值影響而使得平均值的可信度降低。
- (2) 無法明確表示出一首音樂的特性，例如，一首音樂音量，如果以平均來看，那漸升漸降或平穩的音樂將分不出差異。
- (3) 藉由平均得到的音樂特性不易理解，例如，我們說一首音樂的平均音長是 1.25 秒，平均音量是 75 分貝，對沒有豐富專業知識的人來說，無法理解平均音長 1.25 秒是長是短？而平均音量 75 分貝又會對人的心理造成什麼樣的影響，是屬於噪音嗎？

所以我們必須建立出一種新的音樂表示法，以期能更明確表現出音樂的特性，因此，我們利用統計分布情況的方法來表示音樂的特徵，例如，我們會說一首音樂有 80% 的音其音量都是屬於比較大聲的，這樣不但能避免被一些較大或較小的值影響，而且也能很明確地告訴我們這首音樂是屬於比較大聲的。而為了得到特徵的分布情況來釐清對於每一種特徵而言什麼樣的值算大、什麼樣的值算小，我們收集了大量的資料以求得各種特徵所有可能的出現情形 (domain value)，再以 K-Means (K = 3) 的方式將每一種特徵分為大、中、小三類 (L、M、S)，如此一來便可將數值資料轉換為類別資料。因此，一首音樂我們可以用一個 vector 來表

示。

#### 步驟二：自動賦予音樂特徵權重

由於單純使用平均的方式得到的 vector  $M_a$  並不能很正確地表示出使用者的喜好，因此，我們將 vector  $M_a$  加上權重，這是由於每個特徵在不同的情緒／情境中重要度不同，例如，在 happy 的音樂中，聲音大小是一個很重要的判斷要素，而在 peaceful 的音樂中聲音大小的鑑別度可能就不是這麼大，相反地，peaceful 的音樂擁有節奏慢、穩、旋律輕鬆等特性，因此自動調整每個特徵的權重將可以更正確地表示出使用者的喜好情形。而調整權重的方法主要是根據音樂特徵的分布情況來決定，例如，在 happy 中音量有 80% 出現在類別 L，有 15% 出現在類別 M，5% 出現在類別 S；而音高出現在類別 L、M、S 的機率分別為 30%、35%、35%，此時我們可以很清楚地知道對 happy 的音樂而言音量的鑑別度要比音高來得高，所以我們可以根據鑑別度高低給予不同的權重。

#### 步驟三：個人化的情緒／情境音樂檢索

因每個使用者過去經驗的不同對音樂的感受也會不同，所以我們必須根據使用者的回饋來找出符合使用者需求的音樂具備有哪些特徵，因次每回合我們都會根據使用者勾選的音樂計算每個特徵的鑑別度高低。

為了有效利用 non-relevant 的資訊，並更加接近使用者的需求，我們多加入了一個特徵---音色。在聲音物理特性（頻率與震幅）上，影響人類生理及心理情緒因素的三個向量即為：音量 (Loudness)、音高 (Pitch) 與音色 (Timbre) [17]。而在幼兒聽力方面發展順序為音量、音色、旋律、節奏最後方為和聲 [18]。由此可見音色對使用者的喜好影響也是相當大的。

#### 步驟四：相似度計算

相似度的計算使用 Cosine measure，我們可以根據使用者的回饋建立音樂特性 vector  $M_a$  與樂器 vector  $T_f$ ，而後與資料庫中的音樂資料作相似度的計算，因此相似度可以表示成  $\alpha * \text{Sim}_M(M_a, M_n) + \beta * \text{Sim}_T(T_f, T_l)$ 。

## 影像查詢技術之研究

為了達到快速搜尋影像中的物件，我們有下列的步驟：

### 步驟一：建立影像物件的表示方法

由於影像中的物件，常常具有多個屬性，例如：一輛車子具有方向，速度及加速度。因此，我們使用如下的多值序列來表示一個物件的多個屬性：

```
NS1 H H H M M L L
NS NS2 P Z N Z Z N N
NS3 N E E E S W S
```

圖 1：多屬性表示法

### 步驟二：建立查詢模型

由於查詢一個物件時，使用者很難記住該物件的所有屬性，例如，在上個例子中，使用者可能只記住了NS<sub>1</sub>和NS<sub>3</sub>兩個屬性。因此，我們必須允許使用者只針對部分的屬性作查詢。依據上面的例子，一個查詢如圖 2 所示。其中，由於第三個多值屬性與第二個多值屬性都相同，這代表物件沒有改變屬性，因此可以被刪除掉。

```
NS NS1 H H H M M L L
NS3 N E E E S W S
```

圖 2：查詢表示法

### 步驟三：建立索引結構及查詢方法

為了將多值屬性建立索引，我們使用了如圖 3 所示的 KP suffix tree。但是由於 KP suffix tree 只能紀錄單一數值，因此我們將多值屬性拆成好幾條單一數值的屬性，使其能被記載於 KP suffix tree 中。

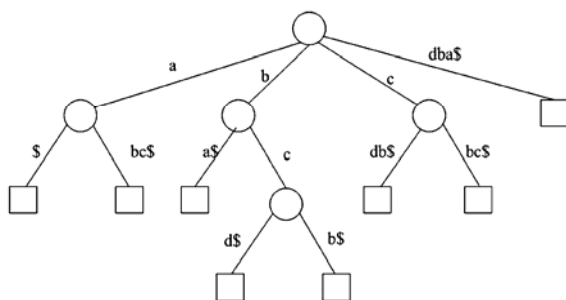


圖 3：索引結構 --- KP suffix tree

而我們在做查詢時，可以利用 KP suffix tree 來找到各個單一數值序列，在判斷其是否有可能組成查詢所要的多值序列。如果可以，則將其組合並去資料庫進行比對。

## 時序性資料走勢之分析及預測

在此，我們以股票作為時序性資料的例子，當然，在做一些修改之後，此技術亦可推廣到其他的應用。參照計劃書中所描述，我們將一股票股價的連續變化，考量不同的區間長度（我們只考量一股票當天的交易情形，因此，區間長度不跨過一天）及多種的技術分析法則（例如，MACD、KD 等技術分析工具），我們將可以從一股票價格的連續變化中，轉換成一多維的事件序列（請參考計劃書頁 31 圖 2）。透過對一股票多維事件序列的分析，我們提出了三個線上推導多維序列片段法則的方法，其中多維序列片段法則中的後項延遲時間，在此定義如同[1]，指前項片段開始時間與後項開始時間的時間差。

首先介紹數個參數，之後再簡述我們所發展的方法。

(1)最小支持度  $min\_sup$ ：是指在  $N$  個單位時間的輸入之後，片段出現次數必須要大於  $(N * min\_sup)$  我們才將該視為頻繁的片段。

(2)最小信賴度  $min\_conf$ ：如果我們發現了一個序列性的片段規則，如：甲 $\rightarrow$ 乙（延遲 15 秒鐘），在此片段規則，只有甲之開始後 15 秒出現乙的次數比上甲出現的次數大於  $min\_conf$  時，我們才輸出此片段法則。

(3)最大延遲限制  $Lmax$ ：我們所求的片段法則，其後項的延遲限制。

定義了上面三種輸入參數後，我們發展了三個方法：

### 方法一：字首樹 (prefix tree) 與時間標籤

一棵字首樹把序列性的片段組織起來。一開始，字首樹僅有一個根結點不包含任何東西。新的項目 A 進來，由最深的節點開始走訪直至根節點。

如果該節點是根節點，便產生一個子節點接在根結點上，這個節點代表一個片段，僅包含一個項目 A。並且在該子節點上記錄這個片段的開始與結束時間。



如果該節點不是根結點，而是一個代表片段乙的節點，首先看現在時間與該片段乙最後兩次所出現的時間，如果某次的開始時間與現在相同，我們便可產生一個子節點，表示一個新片段丙是片段乙的最後一個項目會和項目甲同時出現。如圖 4 及圖 5 所示。

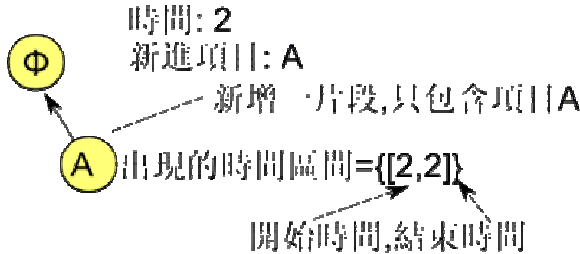


圖 4. 新增單一項目之片段

若其開始時間與現在時間差小於時間限制  $T$ ，表示該次片段乙的出現與現在項目甲的是一個片段丁的出現，在此片段丁是指，片段乙後面接著一個項目 B。

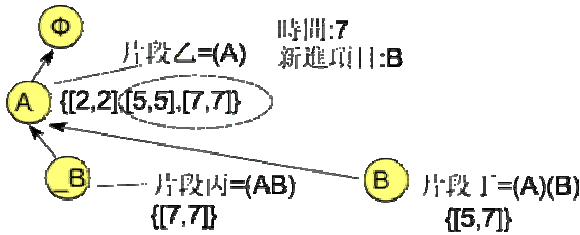


圖 5. 新增兩種新片段

為了要減緩字首樹不停地增長，我們採取了 [11] 所提的失真計算 (Lossy Counting) 的方法，就是每當輸入了固定的資料量之後，便將計算好次數通通減一。我們設定一個誤差值  $\epsilon$  小於最小的出現比例，每當處理過  $1/\epsilon$  個項目組時，我們開始維護這個字首樹，我們將每種片段最早的出現區間去除。如果該節點並未任何出現記錄，我們便把該節點與子樹通通刪除。如圖 6 所示。

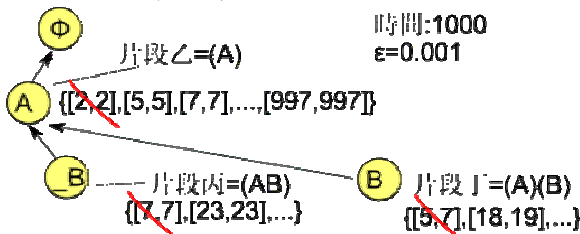


圖 6.  $\epsilon = 0.001$ , 時間 1000 時所有片段的時間標籤去掉一最早的一組

然而在我們的問題中，設定  $\epsilon$  不是那

麼簡單，因為  $\epsilon$  不止影響我們計算片段出現的次數，還會影響到我們計算片段規則的信賴度，經過理論推導，失真計算可能會造成在信賴度上  $\epsilon / \min\_sup$  的相對誤差，所以合理的  $\epsilon$  值約為  $\min\_sup$  的十分之一。

最後對任兩個片段，我們可以拿出他們出現的時間標籤，求出符合條件的片段規則。

方法二：字首樹與延遲資訊

在上述的方法中，每個節點必須要記錄很長的時間標籤，此方法將時間標籤改成記錄延遲資訊。

在每個節點，現在只留下了最後兩組出現的時間區間，以及小於最大延遲限制的開始時間，和一個計算出現次數的計數器，計算片段的方式跟上面一樣，但是處理項目 B (見圖 7) 時，對於一個表示片段乙的節點，我們會先求現在時間與片段乙記錄的小於最大延遲限制的開始時間之差，這個差，便是項目 B 對於片段乙的延遲。我們產生片段丁的節點表示片段乙之後出現項目 B，這些資訊記錄就是在片段丁的節點上。

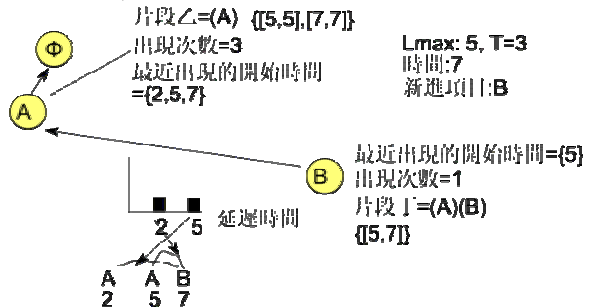


圖 7. 字首樹與延遲資訊

要輸出片段規則，我們走訪過所有結點，將延遲時間次數比上父節點所代表的片段的次數大於最小信賴度，我們便可以輸出這些規則，這些片段規則的後項是只有單一項目。我們可以將這些規則經過組合後，產生出複雜的後項。如：我們發現  $\langle (AB) \rightarrow C \text{ (延遲 5)} \rangle$ ， $\langle (AB) \rightarrow D \text{ (延遲 6)} \rangle$ ，我們便可產生  $\langle (AB) \rightarrow (C)(D) \text{ (延遲 5)} \rangle$ ，我們可以藉由字首樹上記錄的  $\langle (C)(D) \text{ 出現次數} \rangle$  來估計片段規則： $\langle (AB) \rightarrow (C)(D) \text{ (延遲 5)} \rangle$  是否有夠高的信賴度。

### 方法三：字首樹與同時發生表格

最後這個方法，是在系統資源較為不足時可以採用，我們維護  $(Lmax+T-1)$  張表格，每一張表格的欄與列是經由失真計算之後，所滿足  $(min\_sup - \epsilon)$  的項目，除此之外還包含有一個變數  $Lag \cdot Lag$  為  $i$  的表格，在列為項目 A，欄為項目 B 的位置所記錄的值我們記為  $Count(A \rightarrow B, i)$ ，是指 A 出現後  $i$  個單位時間 B 出現的這種情形的次數。如圖 8。

除了每個項目組進來我們都維護這些表格外，我們還是建了一個字首樹，該樹的誤差值是  $min\_sup$ ，並且不記錄時間標籤與延遲資訊。

用這些表格我們可以求出任兩個項目甲乙，所產生的片段規則甲  $\rightarrow$  乙的信賴度。

股票序列: **ABACACB**

lag=1	ABC	lag=2	ABC
A	012	A	210
B	100	B	001
C	110	C	001

圖 8. 方法三所使用的表格(Lag=1, Lag=2 的例子)

因此最後我可以對任兩個字首樹上的節點，估計出在不同延遲，他們所產生的片段規則的信賴度。如圖 9 所示。

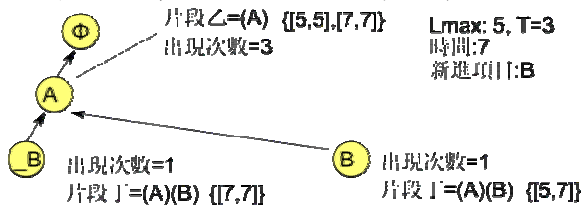


圖 9. 法三所使用的字首樹誤差值是  $min\_sup$ ，並且不記錄時間標籤與延遲資訊

如圖 10，有項目 A、B、C、D，我們想知道  $\langle(AB)(C) \rightarrow (D) \text{ (延遲 } i)\rangle$ ，我們便可以由字首樹知道  $(AB)(C)$  可能發生的次數，再查表格知  $Count(A \rightarrow D, i)$ ， $Count(B \rightarrow D, i)$ ， $Count(C \rightarrow D, i-1)$  的出現次數，再加上下列公式的值：

$$\sum_Y \min\{Count(A \rightarrow Y), Count(Y \rightarrow B)\}$$

取這些值的最小值，便可當成  $\langle(AB)(C) \rightarrow (D) \text{ (延遲 } i)\rangle$  這個規則出現次

數的估計，因此我們便可估計出這些規則的信賴度。

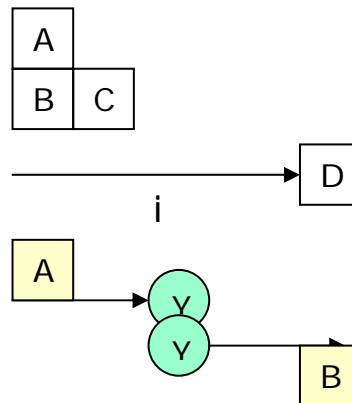


圖 10. 法三中估計片段規則  $\langle(AB)(C) \rightarrow (D) \text{ (延遲 } i)\rangle$  的方法

藉由以上三個方法，我們可以線上快速地處理股票事件序列，並且提供高信賴度的片段法則。

### 五、未來工作

在音樂與影片方面，在計畫的第二年就要進行跨媒體的整合，使得一些自行拍攝的家庭影片在播放時，能自動配合適當的音樂。而在時序性資料的研究方面，我們在第二年希望能依據目前推測的結果，來找出符合推測的序列資料，以達到幫助使用者判斷的目的。

### 六、計劃成果自評

本計劃已達成第一年度的計畫目標，在過去一年裡，我們完成了音樂及影片之特徵選取及分析工作。此外，也在時序性資料的預測方面發展了幾個方法可以互相比較。在音樂資料方面，我們不但完成了音樂的相似搜尋演算法，也同時考慮了音樂所具有的情緒特性。在影片處理方面，我們針對影片物件具有多個屬性的特性，設計了索引以及查詢的方法，使得使用者能依據物件來做搜尋。最後，在時序性資料的預測方面，我們利用關聯規則來達成預測的目的，並且能推斷出事件發生的時間間隔。我們已完成相關論文五篇，其中兩篇已被國際知名會議接受，另外三篇正在準備投稿中。

### 已被接受之論文

- [1] Chia-Han Lin and Arbee L.P. Chen, "Indexing and matching multiple-attribute strings for efficient multimedia query processing," *IEEE Transactions on Multimedia*, 8(2): 408-411, April, 2006.
- [2] Chia-Han Lin and Arbee L.P. Chen, "Approximate Video Search Based on Spatio-Temporal Information of Video Objects," *International Conference on Data Engineering Workshops*, 2006.

### 準備投稿之論文

- [1] Kan Ru Li and Arbee L.P. Chen, "Personalized Music Retrieval Based on Emotions / Situations," *submitted for publication*.
- [2] Jing Wei Liang and Arbee L.P. Chen, "An Efficient and Effective Approach for Music Retrieval," *submitted for publication*.
- [3] Tung-ying Lee and Arbee L.P. Chen, "Mining Serial Episode Rules and Successor Lag Times over Multiple Data Streams," *submitted for publication*.

### 參考文獻

- [1] Emiliios Cambouropoulos, "The Local Boundary Detection Model (LBDM) and its Application in the Study of Expressive Timing," *Proceedings of International Computer Music Conference*, 2001.
- [2] Yazhong Feng, Yueting Zhuang, and Yunhe Pan, "Music Information Retrieval by Detecting Mood via Computational Media Aesthetics," *IEEE/WIC International Conference on Web Intelligence*, 2003.
- [3] Dan Gusfield, "Algorithms on Strings, Trees, and Sequences," *Cambridge University Press*, 1997.
- [4] K.-Y. Huang and C.-H. Chang, "Efficient Mining Strategy for Frequent Serial Episodes in Temporal Database," *The Eighth Asia Pacific Web Conference*, pp. 16-18, 2006.
- [5] Sherri K. Harms, and Jitender S. Deogun,

- "Sequential Association Rule Mining with Time Lags," *Journal of Intelligent Information Systems*, Springer, 22(1): 7-22, 2004.
- [6] Keikki Mannila, Hannu Toivonen, and A. Inkeri Verkamo, "Discovery of Frequent Episodes in Event Sequences," *Data Mining and Knowledge Discovery*, Springer, 1(3): 259-289, 1997.
- [7] J.L. Hsu, C.C Liu and Arbee L.P. Chen, "Discovering Non-trivial Repeating Patterns in Music Data," *IEEE Transactions on Multimedia*, 3(3): 311-325, 2001.
- [8] Dan Liu, Lie Lu, and Hong-Jiang Zhang, "Automatic Mood Detection from Acoustic Music Data," *International Symposium on Music Information Retrieval*, 2003.
- [9] Tao Li and Mitsunori Ogihara, "Detecting emotion in music," *International Symposium on Music Information Retrieval*, 2003.
- [10] N.H. Liu, Y.H. Wu, and Arbee L.P. Chen, "An Efficient Approach to Extracting Approximate Repeating Patterns in Music Databases," *Proceedings of International Conference on Database Systems for Advanced Applications*, pp. 240-252, 2005.
- [11] Gurmeet Singh Manku and Rajeev Motwani, "Approximate frequency counts over data streams," *Proceedings of International Conference on Very Large Data Bases*, pp. 346-357, 2002.
- [12] Pierre-Yves Rolland, "FIEXPAT: Flexible Extraction of Sequential Patterns," *Proceedings of the IEEE International Conference on Data Mining*, pp. 481-488, 2001.
- [13] Hsuan-Huei Shih, Shrikanth S. Narayanan, and C.-C. Jay Kuo, "Automatic Main Melody Extraction From MIDI Files with a Modified Lempel-Ziv Algorithm," *Proceedings of International Symposium on Intelligent Multimedia, Video and Speech Processing*, 2001.
- [14] Stanley Sadie and Alison Latham, 主譯:孟憲福, 審校:殷于涵, "劍橋音樂八

門 (The Cambridge Illustrated Guide of Music),” 果實出版, 城邦文化發行, 2004.

- [15] Alexandra Uitdenbogerd and Justin Zobel, “Melodic Matching Techniques for Large Music Databases,” *Proceedings of the ACM Multimedia Conference*, pp. 57-66, 1999.
- [16] Dan Yang and WonSook Lee, “Disambiguating Music Emotion Using Software Agents,” *International Symposium on Music Information Retrieval*, 2004.
- [17] Zimbardo and G. Philip, 編譯:游恆山, “心理學,” 五南圖書, 1990.
- [18] 王毓雅, “如何進行幼兒音樂教學～由幼兒音樂概念發展觀之,” 國教新知 第47卷第3期