適用於入侵偵測之模糊關聯法則機制
Fuzzy Association Rules Mechanism for Intrusion Detection Systems

曹偉駿　施衣喬
大葉大學資訊管理學系
E-mail: wjtsaur@mail.dyu.edu.tw

摘要

隨著網路使用量越來越高，系統記錄檔中記錄了愈來愈多的資訊，且在駭客入侵系統手法不斷翻新的情況下，如何能正確及有效率的從大量資料中，找出有用的使用者行為樣式並防禦未知型的攻擊手法是入侵偵測系統(Intrusion Detection System)的一項重點。

本研究提出一適用於主機式入侵偵測之模糊關聯法則機制，主要是利用資料探勘的方式產生正常及異常使用者的行為法則，並將這些法則建構於異常及正常的資料庫中，而日後若有新資料產生，則再以模糊漸進式關聯法則對新資料進行運算，以降低產生法則所需運算的時間，並獲得最新的關聯法則。此機制的設計不但能有效率地從大量資料中，精確地建立出使用者行為樣式，在綜合異常及誤用偵測技術後，能提供入侵偵測引擎更精確地判定是否為駭客入侵行為，進而降低錯誤率，並抵擋未知型的入侵手法。而在研究最後，也將開發一套系統驗證本研究提出之機制的成效。

關鍵詞：入侵偵測系統、模糊理論、群集技術、關聯法則、漸進式資料探勘

一、前言

目前大多數入侵偵測系統可分為兩種架構，針對系統記錄檔為樣式比對的稱之為主機式偵測系統；針對網路封包樣式比對的稱之為網路式偵測系統。都是以單一種的偵測技術察覺是否有駭客入侵的行為發生，但單一種偵測技術的入侵偵測系統通常都有誤報率(把正常行為誤報為異常的攻擊)或誤判率(把異常行為誤判為正常行為)過高的情形發生。因此，如何降低偵測系統的錯誤率是極需探討的主題。此外，隨著網路使用量越來越高，在系統記錄檔中所記錄的資料量也非常龐大，我們不但無法從資料的表面上看出來其隱藏的資訊，更無法用人力來分析。雖然簡單的統計方法、電腦報表及資料庫查詢工具可以用來幫助我們分析資料，但是它並不像新的智慧型分析工具功能如此的強大，可以快速且自動找出隱藏及有用的資料。該使用何種方式能正確及有效率的從各種資料庫中，找出使用者的行為樣式也是入侵偵測研究的一個方向。

本研究主要是將模糊理論和資料探勘方法導入入侵偵測系統，以從龐大的系統記錄檔中，有效地發掘出使用者行為樣式，並精確地分類異常使用者行為及正常使用者行為。而本研究也提出一關聯法則建構機制，綜合異常及誤用偵測技術，以對二者缺陷產生互補作用，以降低錯誤率的發生，並提供入侵偵測引擎判定此行為是否為駭客入侵系統之徵兆。在此機制中，所用到的方式包含了異常偵測，因異常偵測模型建立的是正常使用者行為樣式，所以也可以抵擋未知型的入侵手法。

二、相關研究

資訊安全方面的技術，像是認證和存取控制都是以電腦安全為目的所發展出來的，也就是要預防未經認證的入侵者能自由存取和操作他們想要得資訊。若這些預防系統我們稱之為第一道防線，入侵偵測系統就可說是第二道防線，能夠隔離入侵者以防禦電腦系統被入侵

[2]。其中所用到的技術有許多種，以下即是與本研究相關之研究文獻・

2.1 群集技術

將群集技術應用於入侵偵測方面，Marin[14]、Portnoy [15]、Smith [16]等學者都是以 k-means 為基礎的技術，將來源資料分成多個群集，接下來在偵測入侵的行為時，把使用者的各項行為樣式與每群的特徵作比較，以判定其行為是正常或異常，不同的是 Marin [14]在其論文中加入了專家法則，Portnoy [15]以較有效率的方式來分群，而 Smith [16]是將自我組織映射網路 (Self-organiation Maps) 與 k-means 同時應用於資料的分群。但在傳統 k-means 演算法中，會因為初始群集中心的數量及其重心設定的不同，而產生不因此，在 Jiang 等人[10]的論文中提出改良式的 k-means (modified k-means)演算法，並將此演算法應用於電子郵件系統記錄檔，在這些資訊中以網路流量及訊息作自然分群 (unsupervised clustering)，瞭解一般網路行為的習慣，以便能依據此分析結果進而建立網路異常行為的預警系統。

在模糊群集技術應用於入侵偵測方面，Dickerson [4]等人以 FCM 為基礎，建構一個在網路式環境下異常偵測的”模糊入侵辨視引擎”(Fuzzy Intrusion Recognition Engine)，主要目的是要偵測出 DoS 及主機與連接埠掃描的攻擊。

2.2 關聯法則

除了利用群集演算法偵測出異常的使用者行為外，另外也有許多的研究[7, 11,12]採用關聯法則演算法從系統記錄檔中分析使用者行為，並在各種行為特徵中找出其間的關聯，以發掘出可疑事件的關聯性。如 Lee [11,12]等學者利用關聯法則演算法在記錄了系統各個特徵的系統記錄檔中發掘出有用的樣式，這些樣式包括了許多程式和使用者的各項行為，並計算這一系列相關的系統特徵以產生分類器 (classifier)，而這分類器所存的資料就是關聯法則庫，其主要目的是能辨別出各種異常和已知之入侵行為。Hossain [7]利用關聯法則的方式，從資料庫中找出多個特徵值之間的關聯性，之後將關聯法則建構成決策樹型式的分類器，每個定義好的分類器就是使用者的行為模式，用以判定是正常或異常的行為。

在模糊關聯法則應用於入侵偵測方面，Luo [13]等學者利用 Kuok [9]等學者所提出之模糊關聯法則，從網路的流量及系統記錄檔中萃取出有用的資訊，以代表正常使用者的行為樣式。Florez [5]等學者基於 Kuok [9]等學者所提出之模糊關聯法則，改善了支持度及信賴度門檻值的設定，以產生比 Luo [13]等學者更有用的關聯法則，使得從資料庫中挖掘之使用者行為樣式能更正確。

三、適用於入侵偵測之模糊關聯法則機制

3.1 使用者行為建構方式

本研究主要是提出一適用於入侵偵測的機制，此機制基於模糊資料探勘之技術，分析網站伺服器的使用者行為，建構出一般正常使用者行為及具有入侵傾向的使用者行為之關聯法則。因此，本節將詳述此機制中資料分群所採用的群集演算法、模糊關聯法則演算法及本研究提出之漸進式模糊關聯法則演算法。

3.1.1 資料分群

此階段是採用 Tsaur [17]等人提出之改良的群集演算法(Modified mrFCM)，將網站伺服器紀錄檔中的連線資料進行分群的動作。在此階段中，主要是要得出所有資料的群集中心，並計算出每個資料點的隸屬度，以運用於模糊關聯法則的運算。

3.1.2 產生模糊關聯法則

在模糊關聯法則中是利用隸屬度的運算來得到法則的支持度及信賴度，這隸屬度的決定最重要的是找出群集的重心，一旦群集中心求出則整個資料的模糊隸屬度也可以藉由群

集中心建構出來。本研究是利用前述之 Modified mrFCM[17]所計算出的群集中心，做為模糊關聯法則所需隸屬度之依據，再依此隸屬度建構出演算法運算時所需之資料格式。

Hu[8]等人利用 Boolean 演算法(*AND*, *OR* 及 *XOR*) 的方式獲得法則，他們提出了 FGBRMA(Fuzzy Grids Based Rules Mining Algorithm)演算法，利用此種方式只需要對資料庫掃描一次，以降低系統從磁碟讀取資料的時間，但此演算法在利用 *XOR* 運算時，會產生一些無效的法則，如 X_{1k_1} => $X_{2k_1} * X_{3k_1}$。因此本研究中改進 FGBRMA 第二階段獲得法則的方式，以減少無效法則的產生，其符號定義如表 1 所示。

表 1 FGBRMA 符號定義表

符號	定義
FG[*u*]	在第 *u* 列模糊項目的 Boolean 值
FG[*v*]	在第 *v* 列模糊項目的 Boolean 值
FS	包含所有 *k*-維模糊項目的模糊支持度
conf	使用者定義的最小信賴度
FC(*R*)	計算後法則 *R* 之隸屬度

FGBRMA 改進後的演算法如下：

第二階段：產生有效的模糊關聯法則。

若 FG[*u*]不為空集合，則兩個不相同的列 FG[*u*]和 FG[*v*] (*u*＜*v*)，表示為 L_u 和 L_v，個別執行下列步驟：

步驟 1：產生法則的先前(Antecedent)部份。

1-1：讓 *temp* 成為在(FG[*u*] *AND* FG[*v*])中非零元素的數目。

1-2：假如在 FG[*u*] 中非零元素的數目等於 *temp*，則 $L_v \subset L_u$ 保留，並且法則(稱做 *R*)的先前部份，是由 L_u 產生；否則回到步驟 1。

步驟 2：產生法則結果(Consequence)的部份。

2-1：讓 *Con_temp* 成為在(FG[*u*] *XOR* FG[*v*])中非零元素的數目。

2-2：若在 *Con_temp* 中非零元素的數目≧ *temp*+1，則回到步驟 1；否則以(FG[*u*] *XOR* FG[*v*])獲得 *R* 結果的部份。

步驟 3：進行 FC(*R*) = FS(L_v)/FS(L_u)運算，以確認是否要產生法則 *R*，假如 FC(*R*) ≧ *conf*，則 *R* 是有效的，到步驟 4；否則回到步驟 1。

3.1.3 漸進式關聯法則

在經由上述之方法獲得模糊關聯法則後，經過一段時間系統必定會產生新的記錄檔資料，若每當有新增資料的時候，原有的關聯法則就無法正確地反映出資料庫中新的狀況，此時就發生漸進式(incremental)關聯法則更新的問題。最簡單的解決方法，就是對新的資料庫(亦即新資料加舊資料)，重新執行一次資料探勘。不過資料探勘是一個相當耗時的工作，如此的方式實在太浪費計算的時間及硬體的 I/O，所以目前處理法則更新的方法，均利用原有的關聯法則作為基礎，保存舊資料庫中所有的高頻項目，來對新的資料庫進行資料探勘以增進效率。

在目前的研究中，Ayan 等人[1]提出的 UWEP 演算法是以 Apriori 演算法為基礎加以修改，以便能夠利用舊有的高頻項目，來達成快速更新關聯法則的目標。而本研究則依 UWEP(Update With Early Pruning) [1]理論基礎，更進一步地改良此演算法，以進行模糊關聯法則之更新，並應用於前述之 FGBRMA 演算法。UWEP 演算法類似 FUP 演算法[3]，但 UWEP 應用動態向前(look-ahead)的策略，先進行 *PruneSet* 的運算，將不可能成為更新後資料庫 *DB*+*db* 的高頻項目集儘早去除，並預先把會成為高頻項目的元素加入 L_{DB+db}，可快速減少候選項目集的個數。此外，雖然在 UWEP 之後，Lee[6]等人提出了一種 SWF(Sliding Window Filtering)演算法，以分割資料庫的方式提升漸進式探勘的效率，但其方式將會造成 FGBRMA 演算法耗費過多的記憶體，因此本

研究決定以 UWEP 為基礎，改良為模糊漸進式關聯法則的運算，並稱之為 Fuzzy UWEP，其符號定義表如表 2 所示。

表 2 Fuzzy-UWEP 符號定義表

符號	定義
DB	舊的資料庫
db	新增加資料的資料庫
DB+db	*DB*∪ *db* 的資料庫
L_{DB}	*DB* 中所有的高頻模糊項目
L_{DB+db}	*DB+db* 中所有的高頻模糊項目
L^k_A	*A* 資料庫中的高頻 *k*-維模糊項目
minsup	使用者定義的最小支持度
C^k	*DB+db* 中的模糊候選項目集合
$X.support_A$	*A* 資料庫中 *X* 項目的支持度
X.superset	*X* 項目的超集合
PruneSet	在 *DB* 中是高頻，但在 *db* 中支持度為 0 的模糊項目

Fuzzy UWEP 演算法步驟如下：

步驟 1：刪除在 *db* 中支持度=0 的模糊項目，並將剩下的這些模糊項目建立為 1-維模糊候選項目集合(C^1)。

步驟 2：L^1_{DB} 與 C^1 進行 *XOR* 運算，取得 L^1_{DB} 中出現但 C^1 中沒有的項目及在 C^1 出現卻沒有在 L^1_{DB} 中的項目，以建構 *PruneSet*。*PruneSet* = L^1_{DB} *XOR* C^1。

步驟 3：對 *PruneSet* 中的元素進行運算，以步驟 3-1、3-2、3-3 先去除一些不可能成為 *DB+db* 的高頻項目。

3-1：若 *PruneSet* 不為空集合, *X* 為 *PruneSet* 中的第一個元素，否則到步驟 4。

3-2：假如 $X \in L^1_{DB}$，設定 *db* = 0，以公式 (1) 計算 $X.support_{DB+db}$。若 $X.support_{DB+db} >= minsup$，將 *X* 加入 L_{DB+db}，*X.superset* 加入 *PruneSet*，並將 *X.superset* 從中 L_{DB} 移除；若 $X.support_{DB+db} < minsup$，將 *X*、*X.superset* 從 L_{DB} 及 *PruneSet* 中移除。假如 $X \notin L^1_{DB}$，則到步驟 3-3。

$$X.support_{DB+db} = \frac{(X.supportD\ B^* \mid DB \mid) + (X.supportd\ b^* \mid db \mid)}{\mid DB \mid + \mid db \mid} \quad (1)$$

3-3：假如 $X \in C^1$，以公式 (1) 計算 $X.support_{DB+db}$。若 $X.support_{DB+db} >= minsup$，將 *X* 加入 L_{DB+db}，*X* 從 C^1 中移除；若 $X.support_{DB+db} < minsup$，將 *X* 從 C^1 中移除。

3-4：將 *X* 從 *PruneSet* 中移除，並回到步驟 3-1。

步驟 4：*k* = 1。

步驟 5：若 C^k 不為空集合，進行步驟 6；否則停止運算。

步驟 6：進行 L^k_{DB} 中所有元素的運算。以公式 (1) 計算 $X.support_{DB+db}$。若 $X.support_{DB+db} >= minsup$，將 *X* 加入 L_{DB+db}；否則將 *X* 及 *X.superset* 從 L_{DB} 中移除。將 *X* 從 C^k 中移除。

步驟 7：若 *k* >= 2, 進行 C^k 中所有元素的運算。以公式 (1) 計算 $X.support_{DB+db}$。若 $X.support_{DB+db} >= minsup$，將 *X* 加入 L_{DB+db}；否則將 *X* 從 C^k 中移除。

步驟 8：將剩下 L^k_{DB+db} 中所有的元素進行 Join 運算，以產生 C^{k+1}。

步驟 9：設定 *k* = *k* + 1，回到步驟 5。

3.2 模糊關聯法則機制

圖 1 為本研究提出的模糊關聯法則機制架構，此架構適用於入侵偵測系統，其中分為七個部份：

1、 網站記錄檔

在這部份的記錄檔(Log Files)資料，以 Linux 伺服器內的連線記錄為資料來源。

2、 轉換過後的資料

原始的 Log Files 資料中，有些格式不符合本研究之需求。因此，經由數值化的處理，將資料轉換為符合本研究可使用之格式。

3、 分群後的記錄

在此記錄中，主要記錄的是將 Log Files 轉換成可分析的資料後，將這些資料以群集演算法所分出的各個群集。在此本研究將以 Tsaur[17]等學者提出之 Modified mrFCM 為基礎，較精確地將異常及正常的資料分群出來。資料經由群集技術分為各個群集後，由專家判斷每個群集內的資料為正常使用者行為，或是異常使用者行為，並將群集定義為正常群集或異常群集。

4、 群集中心

經由分群演算法得出各個群集後，可得出每個群集的群集中心，而這群集中心將會被記錄下來，以做為新進資料被歸類為正常群集或異常群集的依據。

5、 初次模糊關聯法則運算

在初次模糊關聯法則運算的部分，以 Hu[8]等學者提出之模糊關聯法則為基礎，將先前分群過的正常及異常群集，導出正常及異常行為之關聯法則。建立出這些法則後，可將這些法則再存入法則資料庫。

6、 漸進式模糊關聯法則運算

經由舊有網站記錄檔導出模糊關聯法則後，若以後獲得新的網站記錄檔，則將這些記錄檔轉換後，依先前所存之群集中心將新進資料分類為正常或異常資料。再利用本研究提出之漸進式模糊關聯法則演算法導出新的法則，而不需要再重新運算全部的正常資料及異常資料，以減少系統運算的時間。

7、 法則資料庫

在法則資料庫中儲存的資料，是以模糊關聯法則運算後得出之異常法則及正常法則。這些法則可以提供給專家，做為建構入侵偵測系統之依據。

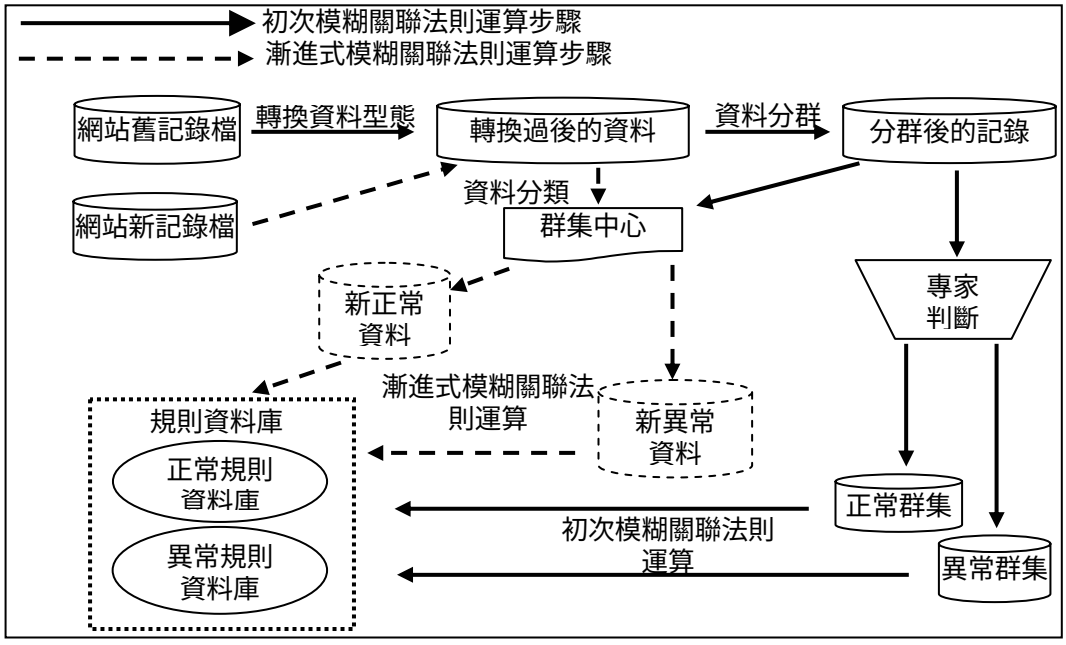

圖 1. 模糊關聯法則機制架構圖

四、實驗分析

4.1 網頁記錄檔分析

在本研究中以校園網路「網頁伺服器系統記錄檔」資料作為分群實驗的對象。我們認為在網頁伺服器系統記錄檔記錄檔中有幾個欄位可為判斷使用者行為法則的依據。因此，我們將記錄檔中含有文字型態的欄位轉換為數值型態，使得本系統能分析這些欄位中所提供的資訊，欄位可為細分成四個部分：

1、 連結時間(稱之為 Time)：

在記錄檔中，每一個連線在幾點鐘建立，如在 01 點建立，相對應的數值資料為 01。

2、 讀取檔案的危險程度(稱之為 Danger)：

讀取目錄，危險程度設定為 2；讀取副檔名為 asp、jsp、js、vbs、pl、cgi、ico 等，危險程度設定為 3；讀取副檔名為 exe、com、dll、ida 等，危險程度設定為 4；讀取危險程度為 3 或 4 的檔案其後再加上參數值，如：/csPassword.cgi?command=remove 以及讀取到系統檔案則危險程度設定為 5；其餘為 htm、html、gif、jpg、bmp、txt 等未列入 2~4 危險程度的讀取檔案方式，則危險程度設定為 1；

3、 Session 計數(稱之為 Session)：

在記錄檔中，每增加一筆資料，對於同一個來源 IP 做計數(連續時間內)，如在 8 個小時內，同一個 IP 對伺服器存取了共 20 次，則 Session 為 20。

4、 狀態代碼(稱之為 Response)：

在記錄檔中會產生的狀態碼可歸類為以下 5 種，其含義為：

100 到 199 代表收到請求，處理這筆連線。

200 到 299 代表成功，這種狀態碼是說明請求已經被成功接受並回應。

300 到 399 代表必須由客戶端採取動作才能滿足所提出的要求。

400 到 499 代表客戶端錯誤，這個狀態碼表示瀏覽器發出的是錯誤的請求。

500 到 599 代表服務端錯誤，這種狀態碼代表服務器回應出現了問題。

因此，在 100 到 199 間，對應代碼為 1；200 到 299 間，對應代碼為 2；300 到 399 間，依此類推。

將記錄檔中的每筆資料的“連結網址”欄位，依照上面四個轉換規則，轉換成一個四維度的特徵向量，如(4, 2, 14, 2)，把所有轉換後的數值資料輸入至程式去分析。

4.2 效率及精確度分析

在本研究中，提出了一 Fuzzy UWEP 模糊漸進式關聯法則演算法，期望能在新資料加入資料庫時，有效地降低模糊關聯法則重新獲得法則的時間。一般模糊關聯法則在新資料加入資料庫運算時，必須重新對新資料及舊資料全部進行運算，而導入 Fuzzy UWEP 後可降低獲得新法則的時間。因此在本研究的實驗中，主要是在驗證前述所提出之 Fuzzy UWEP 演算法之效率及準確性。在整個實證過程中，實驗組為 Fuzzy UWEP 在探勘資料時需花費的時間及獲得法則的資料筆數、內容；對照組為 FGBRMA 在探勘資料時需花費的時間及獲得法則的資料筆數、內容，實證流程如圖 2 所示。為了要驗證 Fuzzy UWEP 之準確性，對照組中的資料和實驗組中的資料內容都相同。

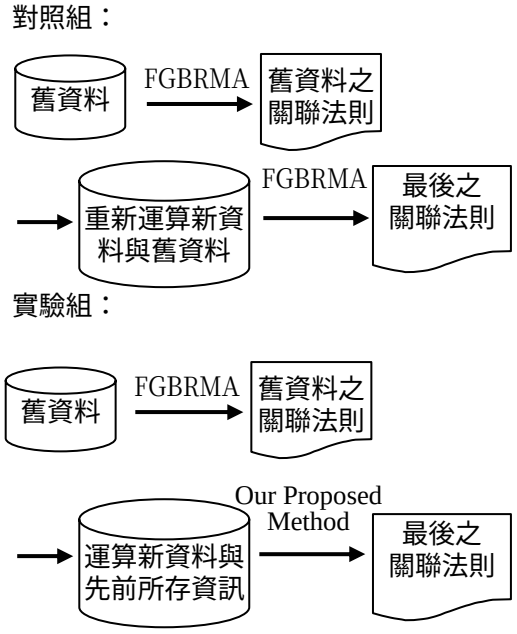

圖 2 實證流程圖

在進行模糊關聯法則運算階段，設定最小支持度為 0.3，最小信賴度為 0.75，實驗的資料筆數為正常群集中任選 100、300、500、700、1000 筆資料。以下為實驗結果的分析：

對照組：對照組中的花費時間為：”在舊資料庫中有 700 筆資料，在新資料庫中有 300 筆資料時，利用 FGBRMA 技術對於總共 1000 筆資料重新探勘”所需花費的時間。表 3 為對照組之實驗數據表。

實驗組：在實驗組中的花費時間為：”在舊資

料庫中有 700 筆資料，在新資料庫中有 300 筆資料時，利用 Fuzzy UWEP 技術對這 1000 筆資料進行探勘”所需花費的時間。表 4 為實驗組之實驗數據表 4。

表 3 對照組數據表

總資料筆數	花費時間(秒)	法則數量
100	1	7 筆
300	4	6 筆
500	6	7 筆
700	11	9 筆
1000	17	10 筆

表 4 實驗組數據表

原始資料筆數	新增資料筆數	花費時間(秒)	法則數量
70	30	1	7 筆
200	100	1	6 筆
300	200	2	7 筆
500	200	3	9 筆
700	300	5	10 筆

經由上述對照組(表 3)及實驗組(表 4)的結果可顯示出，若資料量越大，FGBRMA 則在對資料探勘時，則所需花費的時間也需大量增加；而利用我們提出之方法進行法則探勘時，相對於 FGBRMA 則降低許多時間，如圖 5 所示，X 軸為每次進行探勘的總筆數，Y 軸為每次進行資料探勘所需的時間。另外，在利用我們的方式進行法則探勘後，所得出來新的法則與利用 FGBRMA 重新對全部資料進行運算後，所得出來的法則內容完全相同。由此可知，我們提出之方法確能有效率並準確的對新進資料進行漸進式模糊關聯法則探勘。

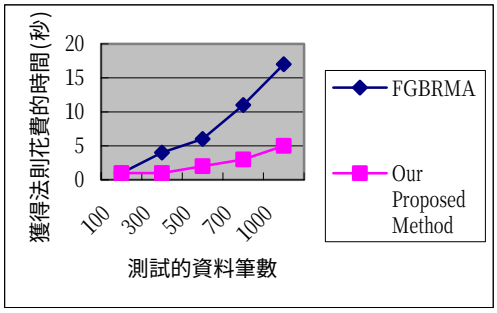

圖 5 法則探勘效率分析圖

五、結論

隨著網路使用量越來越高，暴露在網路上的電腦危險性增加，在系統記錄檔中所記錄的資料也越來越多，資料庫中雖然有很多的資料，但是我們不但無法從表面上看出來其隱藏的資訊，更無法用人力來分析。

因此，本研究提出了一適用於入侵偵測的模糊關聯法則建構機制，是基於 Modified mrFCM 之演算法，有效的將記錄檔中的異常值加以分群出來，讓管理者不需在龐大的記錄檔中去尋找數個異常值。再加上導入模糊關聯法則理論後，對正常及異常群集中的資料進行探勘，建構出每群的使用者行為樣式法則，讓管理者能更加明瞭每筆資料對系統可能造成的危險程度，並加以制定出相對應之防範策略，提供給入侵偵測系統，以降低下次被攻擊入侵之機率。

六、參考文獻

[1] N. Ayan, A. Tansel, and E. Arkun, “An Efficient Algorithm to Update Large Itemsets with Early Pruning, ” *Proceedings of the 5th ACM International Conference on Knowledge Discovery and Data Mining*, pp. 287-291, 1999.

[2] E. Biermann, E. Cloete, and L.M. Venter, “A Comparison of Intrusion Detection systems, Computers and Security,” vol. 20, issue8, pp. 676-683, 2001.

[3] D.W. Cheung, S.D. Lee, and B. Kao, “A

general incremental technique for maintaining discovered association rules," *Proceedings of Database Systems for Advanced Applications*, pp. 185-194, 1997.

[4] J.E. Dickerson, J. Juslin, O. Koukousoula, and J.A. Dickerson, "Fuzzy intrusion detection," *IFSA World Congress and 20th NAFIPS International Conference*, vol. 3, pp. 1506-1510, 2001.

[5] G. Florez, S.A. Bridges, and R.B. Vaughn, "An Improved Algorithm for Fuzzy Data Mining for Intrusion Detection," *Proceedings of the North American Fuzzy Information Processing Society Confere*nce (*NAFIPS- 2002*), pp. 457-462, 2002.

[6] C.H. Lee, C.R. Lin, and M.S. Chen, "Sliding-Window Filtering: An Efficient Algorithm for Incremental Mining," *Proceedings of the ACM 10th International Conference on Information and Knowledge Management*, pp. 263-270, 2001.

[7] M. Hossain, "Integrating Association Rule Mining and Decision Tree Learning for Network Intrusion Detection: A Preliminary Investigation," International Conference on Information Systems, Analysis and Synthesis, vol. 11, pp. 65-70, 2002.

[8] Y.C. Hu, R.S. Chen, G.H. Tzeng, "Discovering fuzzy association rules using fuzzy partition methods," *Knowledge Based Systems*, Vol. 16, pp. 147-147, 2003.

[9] C.M. Kuok, A. Fu, and M. Wong, "Mining Fuzzy Association Rules in Databases," *SIGMOD record*, vol. 17, no.1, pp. 41-46, 1998.

[10] M.F. Jiang, S.S. Tseng, and C.M. Su., "Two-phase clustering process for outliers detection," *Pattern Recognition Letters*, vol. 22, pp. 691-700, 2001.

[11] W. Lee, S.J. Stolfo, and K.W. Mok, "Mining audit data to build intrusion detection models," *In 4th International Conference on Knowledge Discovery and Data Mining*, pp. 66-72, 1998.

[12] W. Lee, S.J. Stolfo and K.W. Mok, "A Data Mining Framework for Building Intrusion Detection Models," *Proceedings of the IEEE Symposium on Security and Privacy*, pp. 120-132, 1999.

[13] J. Luo and Susan M. Bridges, "Mining Fuzzy Association Rules and Fuzzy Frequency Episodes for Intrusion Detection," *International Journal of Intelligent Systems*, vol. 15, pp. 687-703 ,2000.

[14] J.A. Marin, D.J. Ragsdale, and J.R. Surdu, "A Hybrid Approach to Profile Creation and Intrusion Detection," *Proceedings of the DARPA Information Survivability Conference and Exposition - DISCEX* , pp. 69-76 ,2001.

[15] L. Portnoy, E. Eskin, and S.J. Stolfo, "Intrusion Detection with Unlabeled Data Using Clustering," *Proceedings of the ACM CCS Workshop on Data Mining for Security Applications*, 2001.

[16] R. Smith, A. Bivens, M. Embrechts, "Clustering Approaches for Anomaly Based Intrusion Detection," *Walter Lincoln Hawkins Graduate Research Conference*, 2002.

[17] W.J. Tsaur and I.M. Fan, "Anomaly Detection Mechanisms for Web Severs in Linux Environments," *Communications of the CCISA*, vol. 8, no. 4, 2002.