



# Journal of e-Business

第十一卷 第四期 2009年12月 (pp.773~792)

## 結合資料探勘與統計檢定之垃圾郵件過濾器之研究<sup>1</sup>

賴谷鑫<sup>a,\*</sup> 周照偉<sup>b</sup> 陳嘉玫<sup>a</sup>

<sup>a</sup> 國立中山大學資訊管理系 <sup>b</sup> 義守大學資訊管理系

### 摘要

隨著網際網路的普及與電子郵件的廣泛使用，垃圾郵件的數量日益增多，造成電子郵件使用者的不便。當前垃圾郵件相關研究多注重在過濾垃圾郵件之演算法：利用各種人工智慧或是資料探勘的方式來產生垃圾郵件分類的法則，但是隨著垃圾郵件分類法則的長年累積，郵件伺服器可能包含著過時或是無用的垃圾郵件分類法則，而採用過時的法則可能會導致誤判率升高，除此之外，過多的郵件分類法則也會影響郵件伺服器之過濾效能，本研究結合資料探勘與統計檢定的方式做為垃圾郵件防治之完整的解決方案，本研究利用資料探勘的技巧產生垃圾郵件分類法則並且配合統計檢定的方法來決定是否使用此法則對郵件做分類。透過統計模式推導，可以保證所有採用的法則都是高精確度以及高穩定度的法則，如此可以增加垃圾郵件過濾之效能與效率。

關鍵詞：垃圾郵件、資料探勘、統計檢定

## Anti-Spam Filter Based on Data Mining and Statistical Test

Gu-Hsin Lai<sup>a</sup> Chao-Wei Chou<sup>b</sup> Chia-Mei Chen<sup>a</sup>

<sup>a</sup>Department of Information Management, National Sun Yat-Sen University

<sup>b</sup>Department of Information Management, I-Shou University

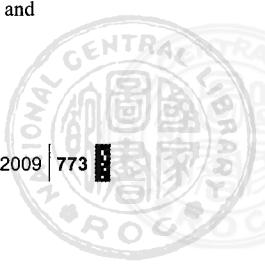
### Abstract

Because of the popularity of Internet and wide use of E-mail the volume of spam mails keeps growing rapidly. The growing volume of spam mails annoys people and affects work efficiency significantly. Most previous researches focused on developing spam filtering algorithm, using statistic or data mining approach to develop precise spam rules. However, mail servers may generate new spam rules constantly and mail server will then carry a growing number of spam rules. The rules might be out-of-date or imprecise to classification as spam

<sup>1</sup> This work was supported in part by TWISCCU, National Science Council under the Grants NSC 97-2219-E-006-009 and NSC97-2218-E-110-004

\* 通訊作者

電子郵件：agl@cm.nsysu.edu.tw





evolves continuously and hence applying such rules might cause misclassification. In addition, too many rules in mail server may affect the performance of mail filters. In this research, we propose an anti-spam approach combining both data mining and statistic test approach. We adopt data mining to generate spam rules and statistic test to evaluate the efficiency of them. By the efficiency of spam rules, only significant rules will be used to classify emails and the rest of rules can be eliminated then for performance improvement.

*Key Words : Spam mail, Data Mining, Statistical Test*

## 1. 前言

垃圾郵件（spam）最簡單的定義是指電子郵件使用者不想收到的信，其內容通常是商業廣告信件、色情信件或是惡意信件。垃圾郵件不管是對個人或是對企業影響都很大，對個人而言，需要花更多時間去過濾信件並且有誤刪重要信件的可能性。而目前很多的惡意程式碼或是網路詐欺與網路釣魚大多透過 spam 方式運作，spam 使得一般人更容易遭受惡意程式碼攻擊或是受到網路釣魚的詐騙；對企業影響方面，除了增加網管人員負擔，減低員工生產力外，其網路釣魚也對企業所造成很大的影響。根據美國 Nucleus Research 於 2007 年四月的調查，spam 造成美國員工生產力的損失，換算起來平均一名員工一年要損失等值於 712 美金的生產力，而整個美國企業因為 spam 所損失的金額為 700 億美金。就以上這些數據可得知，spam 除了造成個人的生活不便之外，對企業影響尤其之大，因此 spam 問題已經是企業刻不容緩需要解決的問題。

目前大部分對於 spam 防治的研究大多著重在利用郵件過濾的方式來進行，然而目前垃圾郵件分類器最大的問題在於：使用者對垃圾郵件的定義不一，例如『徵求論文』的郵件對於學者而言是極為重要的資訊，可是對於上班族而言類似的信件是為垃圾郵件，這也是目前垃圾郵件防治所面臨的首要問題，而本研究認為，要有效防治垃圾郵件必須用多層式的郵件過濾器。一個有效的垃圾郵件過濾器分為兩個層次，首先是伺服器層級的郵件過濾器，此郵件過濾器主要的功能是把『無爭議性』之垃圾郵件過濾掉，此部分郵件可能為帶有惡意碼的郵件、網路釣魚、網路詐欺或是色情郵件，而在用戶端層級的過濾器主要為根據使用者的習慣對垃圾郵件做分類，兩種層級的分類器設計方式差異十分大，而本研究著重在伺服器端的垃圾郵件防治。

伺服器端的垃圾郵件主要目標是堵絕所有無爭議性的垃圾郵件，而目前大多數研究所提出的過濾器方法並不合適，原因在正常郵件誤判率太高，目前大部分垃圾郵件過濾器研究正常郵件誤判率（將正常郵件歸類為垃圾郵件的比率）大約為 1% (Carreras





and Marquez, 2001; Zhao and Zhang, 2005; Zhao and Zhu, 2005）。以使用者角度來看，假設一個人一天收信約 100 封信，那一天可能會有一到四封的正常信件被歸類到垃圾郵件，如此會使得使用者遺失重要的資訊而不自知，因此目前的研究大都不適合在伺服器上的垃圾郵件過濾器；而垃圾郵件過濾的另外一個問題為誤判的成本，就實務上而言，將正常信件歸類為垃圾郵件與將垃圾郵件歸類到正常郵件所付出的誤判成本完全不同，前者的誤判成本遠大於後者。因此需要有一個考慮到不同誤判成本之法則評估的準則。除此之外，目前的研究多著重在法則的產生與郵件的過濾，而少有研究探討垃圾郵件法則的管理，目前垃圾郵件法則的管理缺乏一套具有理論基礎的自動化管理方式。目前管理者管理法則的方式，通常是利用手動去啟用或是停用某特定法則，由於沒有理論上的支持，因此為了避免將正常郵件歸類到垃圾郵件，管理者通常會利用手動的方式放寬過濾標準。而如此一來，垃圾郵件過濾的功能會大大降低。但是當管理者啟用大部分的垃圾郵件法則時候，會因為採用到過多不好的法則造成誤判率增加。隨著垃圾郵件法則的增加，要管理者去監控調整每一個法則的參數是不可能的，因此建立一套具有理論基礎之自動化的法則啟用與停用的機制對於管理者而言是十分重要的。

因此本研究提出一個結合資料探勘與統計檢定的方式做為垃圾郵件之防治，郵件伺服器利用資料探勘的方式不斷的學習新的法則，而當有新的郵件進來，郵件伺服器就會尋找相對應的法則。郵件伺服器會利用統計檢定的方式來對於法則作檢定，唯有檢定後顯著的法則會被用來過濾垃圾郵件，而使用者在收到郵件後可以根據分類結果回饋給郵件伺服器以強化法則的準確度。本研究藉由統計檢定與使用者回饋的方式配合上資料探勘的方法藉以解決伺服器端的垃圾郵件過濾的問題。本論文於第二節探討有關於垃圾郵件之文獻，第三節探討垃圾郵件之法則產生，第四節描述垃圾郵件法則之管理與採用模式，第五節證明本論文所提出的法則管理模式，第六節為展示系統驗證，第七節為結論以及未來研究方向。

## 2. 文獻探討

近年來，垃圾郵件問題日益嚴重，許多科學家紛紛投入垃圾郵件的防治研究，縱覽各垃圾郵件相關文獻，filter 才是最實際的解決方案，表 1 整理了有關利用 filter 作為 spam 過濾之研究。

由表 1 可知，目前對於垃圾郵件的研究，大部分研究多探討垃圾郵件法則的產生，少有研究探討垃圾郵件法則的管理與使用政策，因此本研究提出一個分層過濾垃圾郵件的架構，對於伺服器端的垃圾郵件法則的產生使用與管理，提出了一個完整的解決方案，以有效的減少垃圾郵件的產生。





▼ 表1 研究整理列表

作者	技術
Delany et al. (2005)	利用 Case-Based Reasoning (CBR) 的方式學習垃圾郵件的樣板，藉以過濾垃圾郵件，實驗顯示他的表現結果比 naïve Bayesian 還要好。
Woitaszek et al. (2003)	利用 support vector machine (SVM) 的文字分類的概念，藉由 training 來得到郵件的分類，本研究將系統整合到 Microsoft Outlook 上。
Drucker et al. (1999)	比較 support vector machine (SVM)、Ripper、Rocchio 和 boosting decision tree 等四種不同的方法對垃圾郵件做分析。得到的結果為，當使用 binary feature 時候，SVM 有最好的表現，而 SVM 跟 boosting decision tree 都有著可以接受的結果，只是 SVM 的訓練時間較短。
Bass et al. (1997)	描述遭受到垃圾郵件攻擊場景，並且利用 Perl 根據信件標頭資訊寫出簡單的 filter，除此之外還提出一個簡單的 SMTP Server 架構。
Clark et al. (2003)	利用類神經網路的概念，設計出一個自動郵件過濾器 (filter)，其中作法為將每一個文件利用向量方式表達 (vector)，再經由類神經網路訓練，以自動過濾垃圾郵件。
Robinson (2003)	利用統計的方式 (Bayesian 處理少出現的字，還有卡方分布將獨立的字結合成聯合分布)，而此方法有利用 Python 實作成系統，並持續改進中。
Gee (2003)	利用 latent semantic indexing 的方式作為 filter 的設計基礎，並經由數據顯示來證明他比以往的方法都好。
Zhao and Zhang (2005)	利用約略及合理論 (Rough Set Theory) 配合關鍵字的分佈來產生過濾垃圾郵件的法則。
Zhao and Zhang (2005)	此研究將郵件分為三個類別：正常郵件、垃圾郵件與可疑的郵件，再利用約略集合決策理論方法產生垃圾郵件法則並且過濾研究。
Li and Huang (2002)	利用 support vector machine (SVM) 的文字分類的概念，對於郵件做分類的動作。
Carreras and Marquez (2001)	利用 Boosting Tree 的分析方式對於郵件做分類的動作。
Sahami et al. (1998)	此研究利用 naïve bayesian 的方式來產生垃圾郵件法則，此研究除了利用關鍵字外，也用到標頭屬性 (Header) 來過濾郵件。
Androulatsopoulos et al. (2000)	此研究比較利用 naïve bayesian 方法和關鍵字方法來過濾垃圾郵件，結果顯示 naïve bayesian 有比較好的成果。

### 3. 垃圾郵件法則產生

本研究主要在於提出一個具有理論基礎的垃圾郵件法則管理模式以管理藉由資料探勘所產生出之垃圾郵件法則。因此就理論上而言，任何只要可以產生出垃圾郵件法則的方法均可以用來產生垃圾郵件法則，如決策樹、AdaBoosting with Decision Stumps、Naïve Bayes、Ripper、約略集合理論或是 ADTree。而本研究使用的方法為 Pawlak (2002) 所提出的約略集合理論做為產生垃圾郵件法則的方法。目前已經有一些學





者利用約略集合理論來產生垃圾郵件法則並且有不錯的表現（Zhao and Zhang, 2005; Zhao and Zhu, 2005）。因此本研究利用約略集合理論來產生垃圾郵件法則，本節主要說明利用約略集合理論產生出垃圾郵件法則。

要利用約略集合理論分析資料，首先要建立資訊系統（Information System），一個資訊系統是由全域（一個有限的物件集合）與屬性所組成的，我們可以將資訊系統定義成  $S = (U, A)$  其中  $S$  代表資訊系統， $U$  代表全域 ( $U = \{x_1, x_2 \dots x_n\}$ ， $x_n$  在本研究中代表郵件)， $A$  為屬性的集合， $A = \{a_1, a_2 \dots a_n\}$ ， $a_n$  在本研究中代表一個郵件的屬性，（如有無附加檔案等）。本研究採用三種不同種類的屬性去過濾郵件，分別為關鍵字屬性、標頭屬性（Header）與信件格式屬性。關鍵字是目前研究中最常用來過濾郵件的屬性，藉由計算特定的關鍵字如「賺錢」、「盜版」等等關鍵字數量或是分佈來分類垃圾郵件。而標頭屬性則是根據寄件者，來源 IP 或是有無副本等等的資訊來分類垃圾郵件，有些垃圾郵件會從特定的 IP 發出或是有特定的附加檔案等等，因此此類的資訊也十分重要。至於信件格式屬性則是根據信件內容的格式來分類垃圾郵件，例如有沒有提交表格（submit form）或是是否為 multi part 等等，表 2 列出本研究使用的屬性。

**表 2. 郵件屬性**

屬性名稱	屬性描述
From	寄件者名稱與伺服器來源
Reply to	信件回應時是否有回應給指定的接收者
CC	是有無寄件副本
Received	郵件傳遞的路徑
Subject	郵件主旨
Length	郵件長度（幾 bit）
Domain	件者郵件伺服器的網域名稱
Multi part	是否此郵件為 multi-part
Text/Html	此郵件是文字格式或是網頁格式
Hasform	此郵件有無提交表格
Table	此郵件有無表格
Rec_number	此郵件包含多少關鍵字
Encoding	此研究的編碼方式

在約略集合理論中我們定義  $D$  為一個決策表（Decision Table）其中  $D = (U, A, \cup \{d\})$ ， $d$  為一個不屬於  $A$  的屬性，在本研究中  $d = \{spam, non-spam\}$ ，下表 3 為一個簡化的決策表之範例。



▼ 表 3. 決策表範例

$U \setminus A$	Domain	Subject	Content type	decision
x1	yahoo.com	Money	html	Spam
x2	gmail.com	Homework	text	Non-Spam
x3	mail.nsysu.edu.tw	Homework	text	Non-Spam
x4	yahoo.com	Money	html	Spam
x5	mail.nsysu.edu.tw	Homework	html	Spam
x6	gmail.com	Homework	text	Spam
x7	yahoo.com	Money	html	spam

表 3 提供一個簡化決策表範例，本範例有七個郵件與三個屬性，其中另  $V_a$  為屬性 a 的值域，舉例來說  $V_{Subject} = \{\text{Sex, homework, Money}\}$ ，定義完決策表之後，接下來要定義的為物件間難以辨識的關係（Indiscernibility relation）。對於屬性 B( $B \subset A$ )的每個集合，難以辨識的關係  $IND(B)$  可以藉由下面的方式定義：若  $b(x_i) = b(x_j)$  (對於每個)，則兩個物件  $x_i$  與  $x_j$  藉由屬性 B 的集合是難以辨識的，以表 4 為例， $x_4$  與  $x_7$  對於屬性 A 集合（所有屬性）而言是難以辨識的。 $IND(B)$  的等價類（Equivalence class）稱做 B 中的基本集合（elementary set），他代表物件之最小難以辨識的群體，我們將他定義成  $[x_i]_{IND(B)}$ ，表 4 為根據屬性 Domain 與屬性 Subject 將物件分群的範例。

▼ 表 4. 基本集合範例

U/B.	Domain name	Subject
{x1, x4, x7}	yahoo.com	Money
{x2, x6}	gmail.com	Homework
{x3, x5}	mail.nsysu.edu.tw	Homework

建立基本集合後，接下來就必須做資料的分析，約略集合理論的資料分析是基於兩個概念：下界近似與上界近似（the lower approximations and upper approximations），下界近似表示其元素（element）必定屬於該集合；而上界近似代表元素可能屬於該集合。

令  $X$  表示  $U$  的子集合  $X \subset U$ ，在  $X$  與  $B(B \subset A)$  的下界近似表示為  $B_*(X)$  其中  $B_* = \{x_i \in U \mid [x_i]_{IND(B)} \subset X\}$ ；而  $X$  與  $B$  的上界近似表示為  $B^*(X)$  其中  $B^*(X) = \{x_i \in U \mid [x_i]_{IND(B)} \cap X \neq \emptyset\}$ ；而  $BN_b(X) = B^*(X) - B_*(X)$ ，我們稱他為邊界（boundary），在本研究中  $X$  代





表郵件物件而  $B_*(X)$  代表郵件必定屬於垃圾郵件； $B^*(X)$  代表郵件可能屬於垃圾郵件而  $BN_b(X)$  代表郵件屬於正常郵件。

定義下界近似與上界近似後，就可以求得垃圾郵件的決策法則，在約略集合理論中決策法則是以  $\Phi \rightarrow \Psi$  的型式表達，垃圾郵件法則的精確度定義為  $u_b(X) = \frac{card(B_*(X))}{card(B^*(X))}$  其中  $card(B_*(X))$  代表下界近似中郵件的數量； $card(B^*(X))$  代表上界近似中郵件的數量而  $0 \leq u_b(X) \leq 1$ 。

為了求得高品質的垃圾郵件法則，必須對原始的資訊系統做離散化（discretization）與屬性折減動作（attributes reduction），在郵件物件中，很多的屬性可能是連續之數值，如關鍵字數量或是郵件大小（幾 bit 等等），因此必須對此類屬性的值做離散化，本研究採用 Boolean reasoning algorithm 做為離散化的演算法（Skowron and Son, 1999）。而屬性折減主要目的為在維持相同的基本集合情況下將多餘之屬性刪除，而在約略集合理論中屬性折減是屬於 NP-hard 的問題，本研究利用基因演算法來作為屬性折減的演算法（Wrblewski, 1995）。

本節描述了如何利用約略集合理論產生出垃圾郵件過濾法則，而本研究所提出的方法並不限定在使用約略集合理論，亦可以利用其他資料探勘的方式產生垃圾郵件法則，下一節要介紹的為垃圾郵件管理與採用的模式。

#### 4. 垃圾郵件管理與採用模式

本研究利用報酬（Reward）的概念來對垃圾郵件的法則作為評估，在對郵件做分類時，如果分類正確就回得到正報酬反之亦然。而不同的分類行為有不同程度的報酬，舉例而言將正常郵件分類成垃圾郵件與將垃圾郵件分類到正常郵件其所獲得的負報酬是絕對不同的，表 5 為本研究所定義出的四種不同的報酬。

▼ 表 5 報酬表

Judgment		Spam	Non-Spam
Truth			
Spam		$R_{ss}$	$R_{sn}$
Nom-Spam		$R_{ns}$	$R_{nn}$

$R_{ss}$ ：郵件伺服器將正常的郵件分類為正常信件之報酬。

$R_{sn}$ ：郵件伺服器將正常的郵件分類為垃圾信件之報酬。

$R_{ns}$ ：郵件伺服器將垃圾郵件分類為正常信件之報酬。

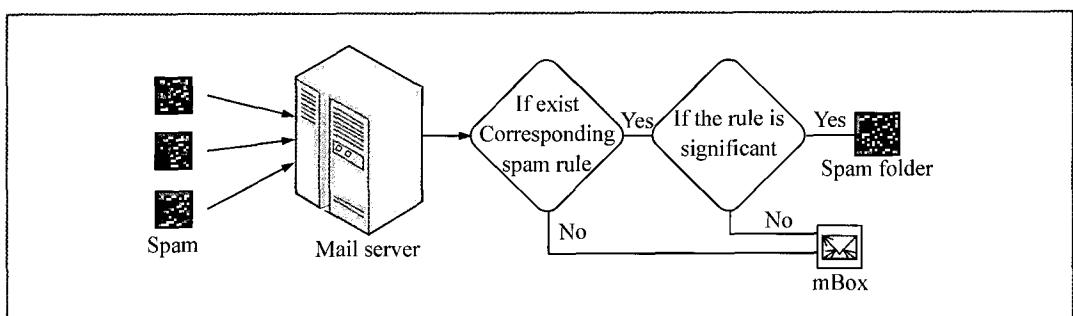
$R_{nn}$ ：郵件伺服器將垃圾郵件分類為垃圾郵件之報酬。



而因為我們研究的為伺服器端的垃圾郵件處理，如果將正常郵件誤判為垃圾郵件成本最大，因此其報酬的大小為下列不等式：

$$R_{ss} > R_{nn} > 0 > R_{sn} > R_{ns} \quad (1)$$

而以上的報酬參數都是可以調整的，可以藉由調整報酬來調整系統效能，本研究主要是利用統計檢定的方式來評估法則的優劣，圖 1 表示本研究提出的方法。



▲ 圖 1 本研究郵件過濾方式

由圖 1 可以了解，以往研究都在於產生精確的垃圾郵件法則，而在法則產生後無條件的（或是用很簡單的條件）去使用他，而這樣的做法沒有考慮到使用者回饋或是法則本身穩定度的問題。本研究利用統計檢定的方式，透過統計檢定來評估運用此法則可能得到的報酬，如果檢定運用此法則所得到的報酬顯著的大於不運用此法則，那郵件伺服器就利用此法則過濾郵件，反之就將此郵件歸類為正常郵件。而根據使用者的不斷回饋，法則的期望報酬會趨近於穩定，因此誤判率就會逐漸降低，下一節將介紹統計模式與證明。

## 5. 統計模式與證明

在描述與證明之前，將之後使用符號作定義，以方便閱讀：

$N$ ：伺服器所有的郵件總數。

$R_i$ ：垃圾郵件法則  $i$ 。

$M_i$ ：郵件  $i$ ， $i = 1, 2, 3, \dots, N$ 。

$P$ ：一封郵件是垃圾郵件的機率。

$\hat{P}_N$ ：伺服器中郵件總數為  $N$  時， $P$  的不偏估計量。

$\beta_i$ ：一封垃圾郵件被  $R_i$  判斷為正常郵件的機率（Type 2 error）。

$\hat{\beta}_i$ ： $\beta_i$  的不偏估計量。





$\alpha_i$ ：一封正常郵件被  $R_i$  判斷為垃圾郵件的機率 (Type 1 error)。

$\hat{\alpha}_i$ ： $\alpha_i$  的不偏估計量。

$\hat{N}_{ss}(i)$ ：利用  $R_i$  將垃圾郵件正確分類的信件數量。

$\hat{N}_{sn}(i)$ ：利用  $R_i$  將垃圾郵件誤判成正常郵件的信件數量。

$\hat{N}_{ns}(i)$ ：利用  $R_i$  將正常郵件誤判成垃圾郵件的信件數量。

$\hat{N}_{nn}(i)$ ：利用  $R_i$  將正常郵件正確分類的信件數量。

$\hat{N}_{ss}$ ：將垃圾郵件正確分類的信件的總數量。

$\hat{N}_{sn}$ ：將垃圾郵件誤判成正常郵件的信件的總數量。

$\hat{N}_{ns}$ ：將正常郵件誤判成垃圾郵件的信件的總數量。

$\hat{N}_{nn}$ ：將正常郵件正確分類的信件的總數量。

$N(i)$ ：利用  $R_i$  做垃圾郵件分類的總郵件數量。

$P(i)$ ：利用  $R_i$  分類的信件中，其信件判為垃圾郵件的機率。

$\hat{P}(i)$ ：為  $P(i)$  的不偏估計量  $\hat{P}(i) = \frac{\hat{N}_{ss}(i) + \hat{N}_{sn}(i)}{N(i)}$ 。

$R(i)$ ：利用  $R_i$  過濾郵件時，所得到的報酬。

而利用  $R_i$  過濾郵件時，所得到的報酬之機率分布近似於常態分布，如方程式(2)所示：

$$R(i) \sim N\left(R_{ss} \cdot P + R_{nn} \cdot (1-P) + P \cdot (R_{sn} - R_{ss}) \cdot \beta_i + (1-p) \cdot (R_{ns} - R_{nn}) \cdot \alpha_i, P^2 \cdot (R_{sn} - R_{ss})^2 \cdot \left[\frac{\beta_i}{N(i)P}\right] + (1-p)^2 \cdot (R_{ns} - R_{nn})^2 \cdot \left[\frac{\alpha_i}{N(i)(1-P)}\right]\right) \quad (2)$$

而如果要檢定兩個法則之間 ( $R_i$  與  $R_j$ ) 之優劣，我們可以先從  $R(i) - R(j)$  之分布著手，如方程式(3)所示：

$$R(i) - R(j) \sim N(\mu_{i-j}, \sigma_{i-j}^2) \quad (3)$$

其中  $\mu_{i-j} = P \cdot (R_{sn} - R_{nn}) \cdot [\beta_i - \beta_j] + (1-P) \cdot (R_{ns} - R_{nn}) \cdot [\alpha_i - \alpha_j]$  (4)

$$\begin{aligned} \sigma_{i-j}^2 &= P^2 \cdot (R_{sn} - R_{ss})^2 \cdot \left[\frac{\beta_i}{N(i)P} - \frac{\beta_j}{N(j)P}\right]^2 + (1-P)^2 \\ &\quad \cdot (R_{ns} - R_{nn})^2 \cdot \left[\frac{\alpha_i}{N(i)(1-P)} - \frac{\alpha_j}{N(j)(1-P)}\right]^2 \end{aligned} \quad (5)$$

$$\mu_{i-j} = P \cdot (R_{sn} - R_{nn}) \cdot [\hat{\beta}_i - \hat{\beta}_j] + (1-P) \cdot (R_{ns} - R_{nn}) \cdot [\hat{\alpha}_i - \hat{\alpha}_j]$$

而要評比兩個法則的優劣可以利用統計檢定的方式，我們假設





$$H_0 : E[R(i)] \leq E[R(j)] \quad (\mu_{i-j} \leq 0)$$

$$H_1 : E[R(i)] > E[R(j)]$$

以  $\hat{\mu}_{i-j}$  為統計量進行檢定，其中

$$\hat{\mu}_{i-j} = P \cdot (R_{sn} - R_{nn}) \cdot [\hat{\beta}_i - \hat{\beta}_j] + (1-P) \cdot (R_{ns} - R_{nn}) \cdot [\hat{\alpha}_i - \hat{\alpha}_j] \quad (5')$$

而如果  $\hat{\mu}_{i-j} > Z_\alpha \cdot \sigma_{i-j}$  我們就拒絕  $H_0$ ，即  $R_i$  優於  $R_j$ ，其中  $Z_\alpha$  為標準常態分布的  $Z$  值。

此外除了兩個法則之間的評比外，系統內部有兩個預設的法則（Non-Spam Rule,  $R_N$  與 All-Spam Rule,  $R_A$ ）：

$R_N$ ：此法則把所有的信件分類為正常郵件，因此  $\alpha_N = 0$ ,  $\beta_N = 1$

$R_A$ ：此法則把所有的信件分類為垃圾郵件，因此  $\alpha_A = 1$ ,  $\beta_A = 0$

而它們的報酬為非隨機變數，分別如下：

$$R(N) = R_m \cdot (1-P) + R_{sn} \cdot P \quad (6)$$

$$R(A) = R_{ns} \cdot (1-P) + R_{ss} \cdot P \quad (7)$$

而當郵件伺服器要利用法則  $R_i$  過濾郵件時，一定要在  $E[R(i)] > R(N)$  且  $E[R(i)] > R(A)$  的狀況下才使用此法則。而我們可以設以下條件成立時才應用法則：

$$\begin{aligned} \hat{\mu}_i - R(N) &> Z_\alpha \cdot \sigma_i \\ \hat{\mu}_i - R(A) &> Z_\alpha \cdot \sigma_i \end{aligned}$$

其中  $\hat{\mu}_i = R_{ss} \cdot P + R_{nn} \cdot (1-P) + P \cdot (R_{sn} - R_{ss}) \cdot \hat{\beta}_i + (1-p) \cdot (R_{ns} - R_{nn}) \cdot \hat{\alpha}_i$

如此我們可以經由統計的推論得知應用此法則後的期望報酬會比不應用好。

以上為本研究的統計模式，以下將要證明上述的統計模式。

首先我們證明 Lemma 1

$$\text{Lemma 1 : } \lim_{N \rightarrow \infty} \Pr[|\hat{P}_N - P| \geq \varepsilon] = 0 \quad \forall \varepsilon > 0$$

首先證明  $\hat{P}_N = \frac{\hat{N}_{ss} + \hat{N}_{sn}}{N}$  為  $P$  的不偏估計量： $\hat{N}_{ss} + \hat{N}_{sn}$  可以寫成信件是否為垃圾郵件之指標函數  $M_i$ ,  $i = 1, 2 \cdots N$  的加總，其中  $M_i$ ,  $i = 1, 2 \cdots N$  為一組符合 i.i.d 的隨機變數，即  $\hat{N}_{ss} + \hat{N}_{sn} = M_1 + M_2 + M_3 + \cdots + M_n$  而  $P(M_i = 1) = 1 - P(M_i = 0)$ ,  $i = 1, 2, \cdots, N$  ( $M_i = 1$  代表此信件為垃圾郵件， $M_i = 0$  代表此信件為正常郵件)。因此  $E(\hat{N}_{ss} + \hat{N}_{sn}) = NP$  且  $E(\hat{P}) = \frac{1}{N}E(\hat{N}_{ss} + \hat{N}_{sn}) = P$ ，故得證  $\hat{P}_N$  為  $P$  的不偏估計量，而根據大數法則得知





$$\lim_{N \rightarrow \infty} \Pr[|\hat{P}_N - P| \geq \varepsilon] = \lim_{N \rightarrow \infty} \Pr\left(\left|\frac{\sum_{i=1}^N M_i}{N} - E\left(\frac{\sum_{i=1}^N M_i}{N}\right)\right| \geq \varepsilon\right) = 0, \forall \varepsilon > 0,$$

因此證得 Lemma 1 為真。

根據 Lemma 1，我們利用  $\hat{P}_N$  取代  $P$ ，而把它視為一個常數，接下來本研究要證明：Lemma 2： $\hat{\alpha}_i = \frac{\hat{N}_{ns}(i)}{\hat{N}_{ns}(i) + \hat{N}_{nn}(i)}$  為  $\alpha_i$  的不偏估計量

證明：在收到的正常信件數量為  $(\hat{N}_{ns}(i) + \hat{N}_{nn}(i))$  之條件下， $\hat{N}_{ns}(i)$  的條件期望值為  $E(\hat{N}_{ns}(i) | \hat{N}_{ns}(i) + \hat{N}_{nn}(i)) = (\hat{N}_{ns}(i) + \hat{N}_{nn}(i)) \cdot \alpha_i$ ，因此  $E\left(\frac{\hat{N}_{ns}(i)}{\hat{N}_{ns}(i) + \hat{N}_{nn}(i)} \mid \hat{N}_{ns}(i) + \hat{N}_{nn}(i)\right) = \alpha_i$ ，利 用 全 機 率 法 則 (Law of Total Probability)  $E\left(\frac{\hat{N}_{ns}(i)}{\hat{N}_{ns}(i) + \hat{N}_{nn}(i)}\right) = E\left[E\left(\frac{\hat{N}_{ns}(i)}{\hat{N}_{ns}(i) + \hat{N}_{nn}(i)} \mid \hat{N}_{ns}(i) + \hat{N}_{nn}(i)\right)\right] = \alpha_i$ ，故得證 Lemma 2。

證明出  $\hat{\alpha}_i$  為  $\alpha_i$  的不偏估計量後，要求得  $\hat{\alpha}_i$  的變異數，而我們在此利用一些近似值的技巧來求得  $\hat{\alpha}_i$  的變異數。

$$Var\left(\frac{\hat{N}_{ns}(i)}{\hat{N}_{ns}(i) + \hat{N}_{nn}(i)}\right) = E\left(\frac{\hat{N}_{ns}(i)}{\hat{N}_{ns}(i) + \hat{N}_{nn}(i)}\right)^2 - E^2\left(\frac{\hat{N}_{ns}(i)}{\hat{N}_{ns}(i) + \hat{N}_{nn}(i)}\right) = E\left(\frac{\hat{N}_{ns}(i)}{\hat{N}_{ns}(i) + \hat{N}_{nn}(i)}\right)^2 - \alpha_i^2$$

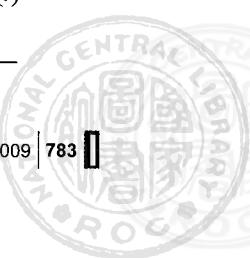
而我們計算或是利用逼近的方式得到  $\hat{\alpha}_i$  與  $E\left(\frac{\hat{N}_{ns}(i)}{\hat{N}_{ns}(i) + \hat{N}_{nn}(i)}\right)^2$  的二階動差。

首先利用  $R_i$  所分類的所有郵件中，是正常郵件的為  $\hat{N}_{ns}(i) + \hat{N}_{nn}(i)$ ，而  $\hat{N}_{ns}(i)$  的條件分佈為二項分佈： $B(\hat{N}_{ns}(i) + \hat{N}_{nn}(i), \alpha_i)$  而其條件二階動差為：

$$E\left[\left(\frac{\hat{N}_{ns}(i)}{\hat{N}_{ns}(i) + \hat{N}_{nn}(i)}\right)^2 \mid \hat{N}_{ns}(i) + \hat{N}_{nn}(i)\right] \cong \frac{\alpha_i}{\hat{N}_{ns}(i) + \hat{N}_{nn}(i)} + \alpha_i^2$$

在此利用逼近的方法將  $E\left(\frac{1}{\hat{N}_{ns}(i) + \hat{N}_{nn}(i)}\right)$  用  $\frac{1}{\hat{N}_{ns}(i) + \hat{N}_{nn}(i)} = \frac{1}{N(i)(1-P)}$  逼近，當  $N(i)$  很大時， $\hat{N}_{ns}(i) + \hat{N}_{nn}(i)$  與其期望值  $E(\hat{N}_{ns}(i) + \hat{N}_{nn}(i))$  之比值接近 1 的機率逼近 1。

根據以上推論及應用中央極限定理，誤判率可以以常態分布來逼近  $\frac{\hat{N}_{ns}(i)}{\hat{N}_{ns}(i) + \hat{N}_{nn}(i)} \sim N\left(\alpha_i, \frac{\alpha_i}{N(1-P)}\right)$  而我們也可以用相同的方式求得  $\hat{\beta}_i$  的分布， $\frac{\hat{N}_{ss}(i)}{\hat{N}_{ss}(i) + \hat{N}_{sn}(i)} \sim N\left(\beta_i, \frac{\beta_i}{NP}\right)$ ，當  $\hat{\beta}_i$  與  $\hat{\alpha}_i$  的分佈求出來後，就可以根據 reward 求得  $R(i)$  的分佈， $R(i)$  也服從常態分佈，其期望值與標準差如下  $R(i) \sim N(R_{ss} \cdot P + R_{nn} \cdot (1-P) + P \cdot (R_{sn} - R_{ss}) \cdot \alpha_i + P \cdot (R_{sn} - R_{ss}) \cdot \beta_i, \sqrt{(R_{ss} - R_{sn})^2 \cdot P \cdot (1-P) + (R_{sn} - R_{ss})^2 \cdot \alpha_i^2 + (R_{sn} - R_{ss})^2 \cdot \beta_i^2})$





$$R_{ss}) \cdot \hat{\beta}_i + (1-p) \cdot (R_{ns} - R_{nn}) \cdot \hat{\alpha}_i, P^2 \cdot (R_{sn} - R_{ss})^2 \cdot \left( \frac{\hat{\beta}_i}{N(i)P} \right) + (1-p)^2 \cdot (R_{ns} - R_{nn})^2 \cdot \left( \frac{\hat{\alpha}_i}{N(i)(1-P)} \right) \text{ 同(2)}$$

得到出法則的分佈後，可以透過它來對法則做檢定，透過不斷的回饋，可以達到自動法則管理的功效。

## 6. 實驗結果與驗證

為了驗證本論文所提出的方法，本論文利用真實資料實際撰寫基於約略集合理論之垃圾郵件法則產生器以及過濾器，實驗組系統有加入法則管理模組而對照組系統則是直接採用約略集合理論所產生出來的法則過濾郵件，本論文將利用下一節介紹之效能分析指標作為驗證數據。

### 6.1 垃圾郵件過濾效能分析指標

本論文用兩種類型的指標作為系統驗證，第一種類型指標為判斷率，本論文利用郵件正確率（Spam precision）、郵件召回率（Spam recall）、精確率（Accuracy）、錯誤率（Miss rate）四項，而整個系統判斷結果可以歸納成表 6。

▼ 表 6：實際郵件與過濾郵件之關係矩陣

實際郵件類別 系統判定類別	實際垃圾郵件	實際正常郵件
系統判定為垃圾郵件	True Positive (TP)	False Positive (FP)
系統判定為正常郵件	False Negative (FN)	True Negative (TN)

整個系統判斷指標計算公式如下：

$$\text{Spam precision} = \text{TP}/(\text{TP} + \text{FP}) \quad (8)$$

$$\text{Spam recall} = \text{TP}/(\text{TP} + \text{FN}) \quad (9)$$

$$\text{Accuracy} = \text{TN} + \text{TP} / (\text{TP} + \text{FP} + \text{FN} + \text{TN}) \quad (10)$$

$$\text{Miss rate} = \text{FP}/(\text{FP} + \text{TN}) \quad (11)$$

而另一個重要的指標為系統累積的報酬作為一量化指標，其系統總期望值為  $\hat{N}_{ss} \cdot R_{ss} + \hat{N}_{sn} \cdot R_{sn} + \hat{N}_{ns} \cdot R_{ns} + \hat{N}_{nn} \cdot R_{nn}$ 。藉由此整體報酬來驗證本論文所提出的模式。



## 6.2 實驗資料

垃圾郵件過濾相關論文中，所實驗採用的資料集可分為真實的資料以及學術單位所提供之資料庫資料。目前公開的資料庫以 UCI Spambase Data Set (UCI, 1999) 以及 TREC Public Spam Corpus (TREC, 2007) 最常被用於垃圾郵件過濾器之系統驗證。而本論文所採用的資料為真實且正在運作之郵件伺服器資料，資料來源為南部某大專院校之郵件伺服器。本論文不採用公開的資料庫原因如下。首先對於 UCI 資料庫而言，此資料庫之資料為處理過之資料且缺乏時間資訊。本論文所提出的方法強調法則管理，使用者回饋，若無時間資訊只靠亂數取訓練資料以及測試資料並不符合本論文假設；TREC 2007 提供完整的郵件資料，而不採用他的理由為：(1)此資料集並非單一伺服器所收集資料，而本論文主要研究為單一伺服器法則管理，資料收集方式並不相同；(2) TREC 2007 為各地收集或是回報的資料，所收集的資料垃圾郵件以及正常郵件的比例與目前真實社會中的比例不符合（本論文：90%，而 TREC 2007：67%），本論文引用統計的方式，如果比例不正確的話會影響最後得到的數據。基於以上理由，本論文決定採用最新收集之郵件伺服器資料，所蒐集的資料來源以及分類如表 7。

▼ 表 7. 實驗資料一覽表

類別	3/1-4/25 郵件數目	垃圾郵件比例
垃圾郵件	54,602	9
正常郵件	5,921	1

本論文實驗以每週為單位，每一週的郵件分成一個資料集，以第一週以及第二週 (3/1~14) 作為訓練資料，代號為 T0 與 T1，以 S0 到 S5 表示每一週的測試資料集；S0 表示第一週測試資料，透過 T0 以及 T1 以所訓練的垃圾郵件法則來過濾 S0 的郵件，並將結果回饋。原始的垃圾郵件法則，就只有 T0 以及 T1 所訓練的資料。但每週持續回饋的垃圾郵件資料，持續強化垃圾郵件法則。例如，第二週後，所使用的垃圾郵件法則，包含原始的垃圾郵件法則，並且加入 S1 的垃圾郵件法則。第三週包含原始法則 S0 的法則，以及 S1 和 S2 的法則。

## 6.4 實驗參數與結果

本實驗所用重要參數列於表 8。

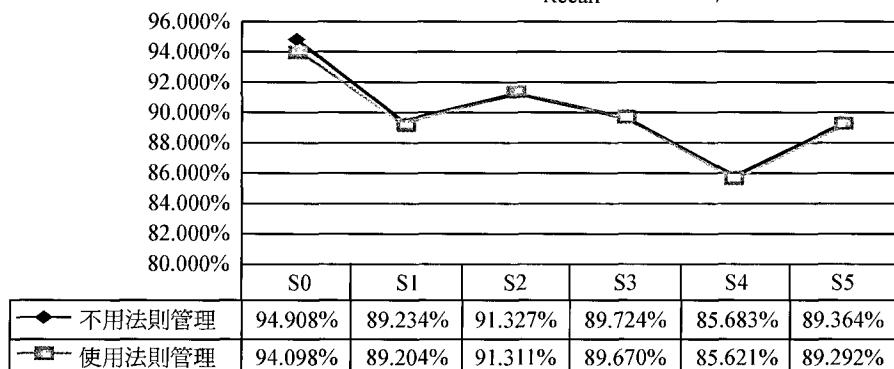
本節將對五項重要指標之數據做一比對以及說明，圖 2 顯示為郵件召回率 (Spam recall) 之比較數據。

藉由上圖可以知道有沒有利用統計模式來做法則管理對於 Spam Recall 影響無顯著差異，而就數字上而言，不用法則管理的表現略佳（但是差異小）。這是因為 Spam

▼ 表 8：實驗重要參數

參數名稱	數值
$R_{ss}$	20
$R_{ns}$	-100
$R_{sn}$	-20
$R_{nn}$	10
統計信賴區間	99%

Recall

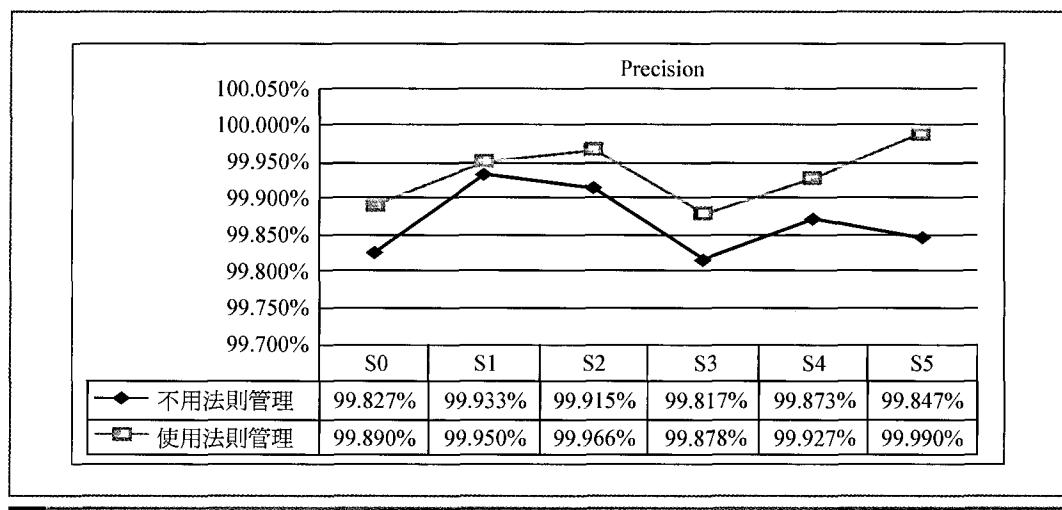


▲ 圖 2：Spam Recall

Recall 本身的意涵為在郵件為垃圾郵件的情況下，可以正確判斷垃圾郵件能力的指標。而如果有利用本論文提出的統計模式為基礎做為法則管理，一開始會因為某些法則不夠顯著而不採用此法則，因此會造成一開始 Spam Recall 較低的情況，而就本實驗結果顯示，實驗組以及對照組的 Spam Recall 並無顯著差異，因此可以證明本論文所提出的方式有與傳統方式相同的垃圾郵件過濾能力。而另外一項垃圾郵件判斷效能的指標為郵件正確率（Spam precision），圖 3 為實驗結果。

由圖 3 可知實驗組與對照組的 Spam Precision 的無顯著差異，而本研究所提出的方法略高於對照組，Spam Precision 本身的意涵為：在被系統判為垃圾郵件的情況下實際為垃圾郵件的比率，也就是說如果 Spam Precision 高，其代表正常的郵件比較不會被判為垃圾郵件。這對於一個垃圾郵件過濾器是一個很重要的指標，而本論文主要的目的也在於在維持一定的判斷率情況下減少正常郵件被判為垃圾郵件的機會。但是在此實驗結果並不顯著，這是因為 Spam Precision 的分母為被系統判為垃圾郵件的數量，

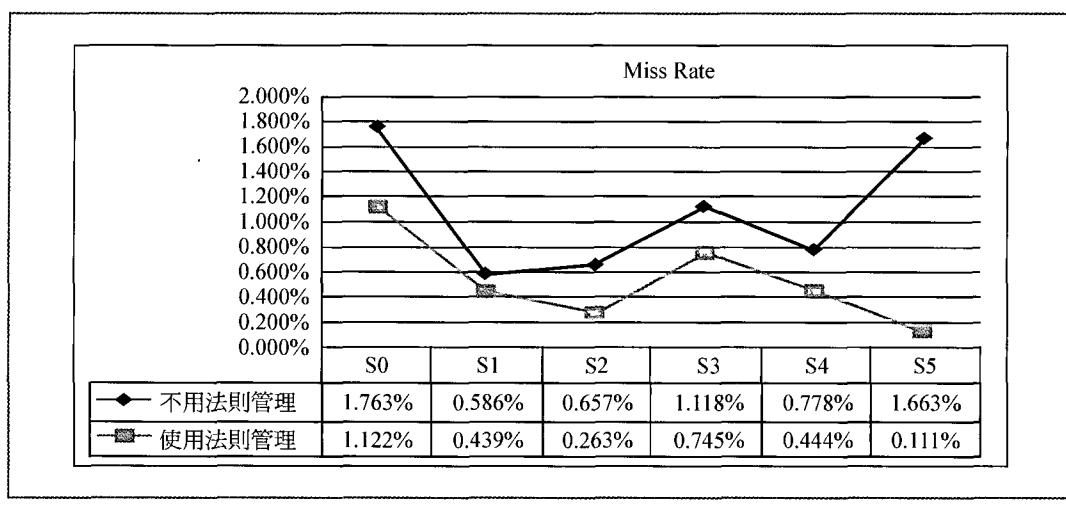




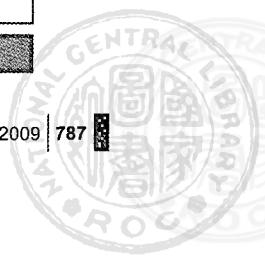
▲ 圖 3 Spam Precision

而分子為真正垃圾郵件的數量。當垃圾郵件的比率以及數量非常大的時候，正常郵件數量相對之下十分少，而正常郵件又被誤判為垃圾郵件之數量更為稀少，因此造成實驗組與對照組比較之下不顯著。有鑑於此本論文利用錯誤率（Miss Rate）作為系統是否會將正常郵件誤判為垃圾郵件之指標，其比較結果如圖 4 所示。

由圖 4 得知，本論文實驗組的 Miss Rate 的確低於對照組，而 Miss Rate 代表著在信件為正常信件的情況下被判斷為垃圾郵件的機會。就正常的系統下，這是一個非常重要的指標，因為 Miss Rate 高的系統比較會讓正常郵件被誤判為垃圾郵件。與 Spam Precision 不同的是，Miss Rate 以正常郵件當分母，因此能夠反映出真實的正常郵件被

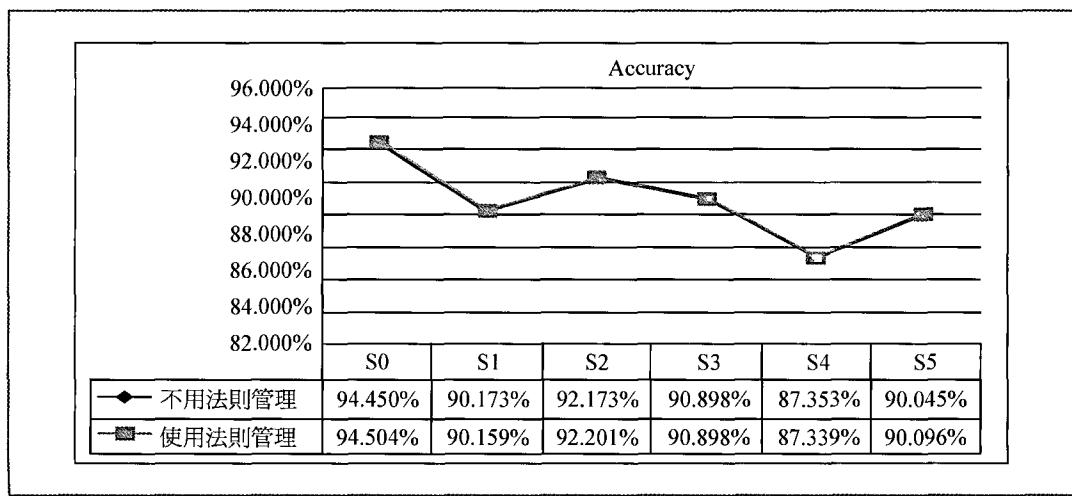


▲ 圖 4 Miss Rate





誤判的情況。經由實驗結果可證明本論文所提出的方法可以有效降低 Miss Rate。而以上三種的指標是分別根據垃圾郵件以及正常郵件的判斷正確率當作指標，而圖 5 所顯示的精確率（Accuracy）代表的為整體系統的判斷率。



▲ 圖 5 Accuracy

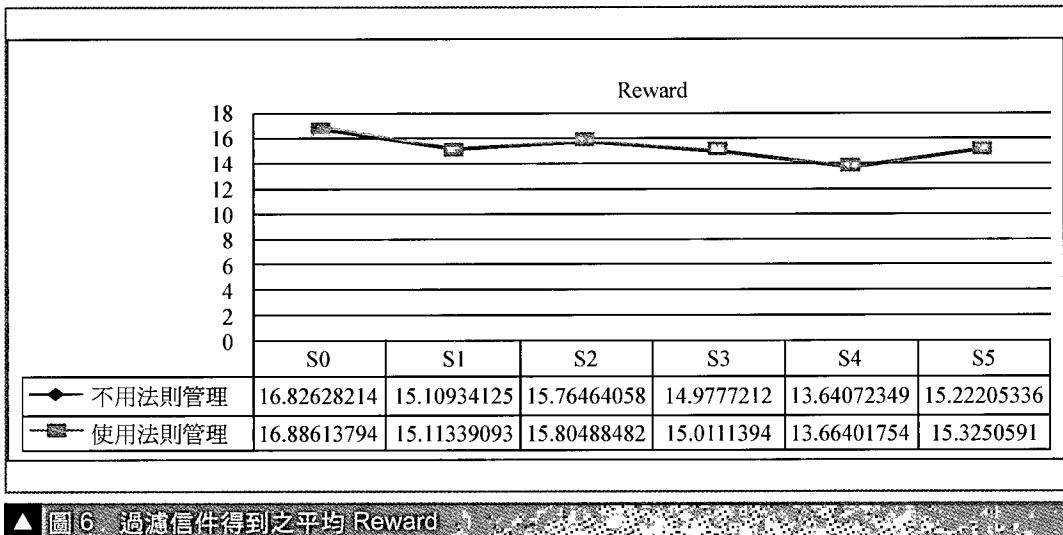
圖 5 顯示本論文所提出的方法整體表現略優於傳統方式，而在垃圾郵件數量以及比率遠高於正常郵件的情況下，要判斷一個垃圾郵件過濾器的好壞主要在於 Miss Rate。這是由於假設垃圾郵件數量十分巨大的情況下 Accuracy、Recall 以及 Precision 本身分母包含有垃圾郵件數量，當垃圾郵件數量一多，就算都不做分類（單純的將所有郵件都視為 Spam mail）其整體表現就數據上而言也不會差很多；反之 Miss Rate 本身的分母為正常郵件，數量比較少，可以很公平的判斷整個系統會不會將正常郵件視為垃圾郵件。實驗結果顯示本論文所提出的方式不但可以對整體的垃圾郵件判斷率有幫助，最重要的是在 Miss Rate 上有顯著的降低。

以上的實驗主要根據垃圾郵件以及正常郵件的判斷率指標驗證本論文所提供的管理模式。而在本論文中，另一個重要的指標為是依照報酬來判斷系統的優劣，因此圖 6 為系統過濾信件，每封信件所獲得之平均報酬之比較。

由圖 6 可以得知本論文所提出的方式每封信可以得到較多的報酬，但是從數據上來看比較之下差異不是很大，這是因為大部分（超過 90%）的信件為垃圾郵件，而只要 Miss Rate 不是差異十分大的情況下系統所得到的 Reward 就顯示不出有十分巨大的差異，不過就實驗數據來看，本論文所提出的方式對於整個系統的判斷率以及得到的回饋確實有所改善。

根據以上的實驗得到以下的結論，本論文所提出的方式可以在小幅度改善整體的過濾效能情況下（不犧牲垃圾郵件之過濾能力）來大幅改善 Miss Rate。而目前對於垃





▲ 圖6. 過濾信件得到之平均 Reward

圾郵件系統的指標，在垃圾郵件比率越來越大的情況下要求大幅度的改善整體的判斷率是無意義的，因為只要將所有的郵件都歸類到垃圾郵件，就有 90%以上的判斷率。而往後的垃圾郵件比率將會不斷增加，因此 Miss Rate 就成為一個重要的指標。而一般要降低 Miss Rate 會將過濾的條件設定比較嚴格，但隨之而來的是整體過濾率以及垃圾郵件判斷率會下降。因此一個好的垃圾郵件過濾系統必須在不犧牲整體判斷率的條件下降低 Miss Rate。而根據實驗顯示，本論文所提供的方式可以在小幅度改善整體的過濾效能情況下，來大幅改善 Miss Rate。而以報酬或是成本的角度來看，實驗證明本論文所提出的方式對系統整體效能有所改進。

## 7. 結論與未來研究

本論文提供在伺服器端解決垃圾郵件之方案，除了利用資料探勘的方式產生垃圾郵件法則外，更提出一個具有統計理論基礎之自動化垃圾郵件法則管理方式來管理日與俱增法則。藉由統計檢定方式可以保證應用此法則的報酬會比不用（或是利用其他的法則）還要好。而本研究的貢獻如下：(1)本研究提供了一個產生垃圾郵件法則的方式，此方式除了使用關鍵字的分佈外，更加上了格式與表頭的資訊；(2)垃圾郵件法則的管理可以使得不穩定，不精確，不常用的法則不會被用來過濾垃圾郵件，藉以降低誤判率；(3)伺服器管理者不需要手動的啟用或是停用法則，有一個具有理論基礎的自動化方式處理法則停用啟用，可以節省管理者負擔；(4)藉由停用不好的法則可以減少郵件伺服器內法則的數量，如此可以增加郵件伺服器過濾的效率。



本論文提出一個有效的法則管理與產生的模式，而未來的研究可以著重伺服器間垃圾郵件資訊交換，藉由結合資訊交換以及法則管理，達到即時預防垃圾郵件的理想。

## 參考文獻

- Androutsopoulos, I., Palouras, G., Karkaletsis, V., Sakkis, G., Spyropoulos, C. D., and Stamatopoulos, P. (2000), "Learning to filter spam e-mail: A comparison of a Naïve Bayesian and a memory-based approach," In *4th PKDD's Workshop on Machine Learning and Textual Information Access*, 1-13.
- Bass, T. and Watt, G. (1997), "A simple framework for filtering queued SMTP mail (cyber-warcountermeasures)," In *Military Communications Conference*, 3, 1140-1144.
- Carreras, X. and Marquez, L. (2001), "Boosting trees for anti-spam email filtering," *4th International Conference on Recent Advances in Natural Language Processing*, 58-64
- Clark, J., Koprinska, I., and Poon, J. (2003), "A neural network based approach to automated e-mail classification," In *IEEE/WIC International Conference on Web Intelligence*, 702 -705.
- Delanya, S. J., Cunningham, P., Tsymbal, A., and Coyle, L. (2005), "A case-based technique for tracking concept drift in spam filtering," *Knowledge-Based Systems*, 18(4-5), 187-195.
- Drucker, H., Wu, D., and Vapnik, V. N. (1999), "Support vector machines for spam categorization," *IEEE Transactions on Neural Networks*, 10(5), 1048-1054.
- Gee, K. R. (2003), "Using latent semantic indexing to filter spam," In *Proceedings of the 2003 ACM symposium on Applied computing*, 460-464.
- Li, K. and Huang, H. (2002), "An architecture of active learning SVMs for spam," In *6th International Conference on Signal Processing*, 2, 1247-1250.
- Nucleus Research (2007), "Nucleus Research: Spam Costing US Businesses \$712 Per Employee Each Year," Retrieved Oct. 2009, from <http://nucleusresearch.com/>
- Pawlak, Z. (2002), "Rough sets and intelligent data analysis," *Information Sciences*, 147 (1-4), 1-12.
- Robision, G. (2003), "A statistical approach to the spam problem," *Linux Journal*, 2003 (107), 3.
- Sahami, M., Dumais, S., Heckerman, D., and Horvitz, E. (1998), "A Bayesian approach to filtering junk e-mail," In *Proceedings of the AAAI-98 Workshop on Learning for Text*





*Categorization*, 55-62

- Skowron, A. and Son, N. (1999), "Boolean reasoning scheme with some applications in data mining," In *Proceedings of the Third European Conference on Principles of Data Mining and Knowledge Discovery*, 107-115.
- Wojtaszek, M., Shaaban, M., and Czernikowski, R. (2003), "Identifying junk electronic mail in Microsoft outlook with a support vector machine," In *2003 Symposium on Applications and the Internet*, 66-169.
- Wrblewski, J. (1995), "Finding minimal reducts using genetic algorithms," In *Proceeding of the Second Annual Joint Conference on Information Sciences*, 186-189.
- Zhao, W. and Zhang, Z. (2005), "An email classification model based on rough set theory," In *Proceedings of the International Conference on Active Media Technology*, 403-408.
- Zhao, W. and Zhu, Y. (2005), "An email classification scheme based on decision-theoretic rough set theory and analysis of email security," In *2005 IEEE TENCON*, 1-6.

