# 適用於網路入侵偵測不平衡資料之階層式多重分類器

張智傑 王勝德
國立臺灣大學電機資訊學院電機工程學系
{r01921025, sdwang}@ntu.edu.tw

## 摘要

網路活動在近幾年行動裝置普及和雲端化趨勢的推動下有顯著成長，因此入侵偵測系統的存在是非常重要的。由於實際網路流量中相對於正常連接，攻擊的存在是少量的，因此許多基於統計模型的監督式入侵偵測系統不易偵測與分類這些少量但有害的攻擊。本研究中，提出一個基於多個分類器的結合並透過階層式分類平衡數據量的入侵偵測系統，依資料中各類的錯誤成本敏感程度與類包含資料的數目作為分割依據，利用多個二元分類器與一個多類分類器將資料中的每一類依序找出。此方法優點在於富彈性適合各種流行的分類演算法，同時不需修改原始訓練資料統計分布，可以降低入侵偵測中因為原始訓練資料集的各類資料數量相差過大造成的分類誤差，對錯誤成本較敏感的網路入侵資料平均成本也有降低。實驗與結果評估採用 KDD CUP 99 資料集入侵偵測資料集以及其修改後之 ND-KDD 資料集測試，在 ND-KDD 資料集實驗，四種演算法使用階層式多重分類器的錯誤率平均降低百分之十六，平均成本降低百分之十三。

**關鍵詞：**入侵偵測系統、不平衡資料集、階層式分類器

## 壹、緒論

近年來，由於智慧型移動裝置數量逐年上漲，提供雲端服務的組織或公司也持續增加，人們與網路的連結越來越緊密，因此網路安全與個人隱私的重要性是不容忽視的。入侵偵測系統是維護網路安全與隱私的一個重要環節，IDS 是一個自動化監控一台主機或一段網路的軟體，分析可能發生的攻擊事件，如某種蠕蟲感染網路、間諜軟體傳播或是攻擊者想要繞過系統取得未授權的訪問權限等等。

依偵測的技術可以將 IDS 分為兩類，誤用偵測(misuse detection)與異常偵測(anomaly detection)。其中誤用偵測需要人工蒐集各類型的攻擊以提取各攻擊的規則作為定義及比對，也稱為簽名(signature)，其優點是可以簡單添加新的攻擊規則，但缺點是無法辨識新型態的未知攻擊，同時隨著規則累積整體系統將會越來越龐大[14]，現有許多成熟且知名的 IDS 軟體或是防毒軟體都有採用這樣的方式偵測，如 snort、bro 等等[5, 23]。而異常偵測則是預先蒐集資料，藉由此資料利用資料探勘或是機器學習的方法產生正常或異常的模型，依模型判斷事件應屬於正常或是歸為異常。

異常偵測可以依偵測模型建立的方式分為三大類[6]：監督異常偵測，利用標記為正常與異常的資料集訓練分類器；半監督異常偵測，利用標記的資料集加上部分為標記的資料作為訓練資料集；無監督異常偵測，則是利用聚類方法或是尋找離群值的方式找出異常，不需要標記過的訓練資料集。此三類方法的訓練與測試評估效果皆需要資料集，而 KDD CUP 99 資料集[1]即為常常使用的資料集。本研究使用的方法皆屬於有監督類異常偵測。

然而，不論在 KDD CUP 99 資料集或是現實的網路流量數據中，標示異常的資料或是實際為異常的連接都只是佔極少數的一部分。如此隱含不平衡的資料分類情況下分出佔小部分的異常極為困難，訓練出的分類器也有較大的誤差。在入侵偵測的情況下，佔少數的攻擊連接常常是危害較大的連接，如 KDD CUP 99 資料集的 Remote to Local(R2L)與 User to Root(U2R)兩類的攻擊數據只佔 0.23%與 0.01%，此兩類攻擊可能造成攻擊者獲取不合法權限。因此，這樣的不平衡數據可能造成較危險的攻擊數據卻不易被偵測的狀況[12]。

本研究中，針對入侵偵測系統提出一個階層式多分類器架構。調整多分類器的排序與架構因應入侵偵測資料集的內部不平衡與入侵偵測分類錯誤成本的需求。此分類架構有以下優點：

1. 不經過無根據的重新取樣或是手動帶偏見的修改訓練資料，對於網路入侵偵測不平衡資料的小類偵測率有非常好的改善效果。
2. 對於融合多種攻擊類型的新型態攻擊，也有較好的提升效果。

集合上述優點對於不平衡的資料集分類，可能可以提升較小類的分類精確度以及有提升總分類精確度的可能性。

本研究將在，第二章介紹基於網路的入侵偵測系統。第三章略述使用的各類預處理及分類演算法。第四章將說明本研究使用的資料集與主要提出的架構及評估函數。第五章則針對實驗結果進行分析並討論。第六章為結論及探討未來研究的方向。

## 貳、相關文獻

在網路入侵偵測(Network Intrusion Detection System，簡稱 NIDS)的領域中做演算法效果評估時，常使用 KDD CUP 99 資料集作為測試資料集，但單一分類器往往遇到一些問題。如 KDD CUP 99 的第二名與第一名相比有較高的總分類精確度，但平均成本表現不夠好，而第一名之所以能夠脫穎而出，是因為能夠透過極端不平衡的資料訓練分類器成功偵測佔極小比例的 R2L 攻擊，同時保持其他分類的正確性[11]。除此之外，基於錯放攻擊產生的成本往往高於將正確的連接視為攻擊的成本，所以 NIDS 的分類方式最好盡可能的抓出攻擊連接，就算犧牲一些正常連接也可以接受。

為了達成以上的效果，在一些利用各式預處理搭配分類演算法的研究中顯示，適當的預處理可以提升演算法的分類效果，在網路入侵偵測情況搭配適當的離散化方法與屬性選擇方法可以提升分類精確度，但未有考慮到資料集的不平衡特性，未有利用多分類

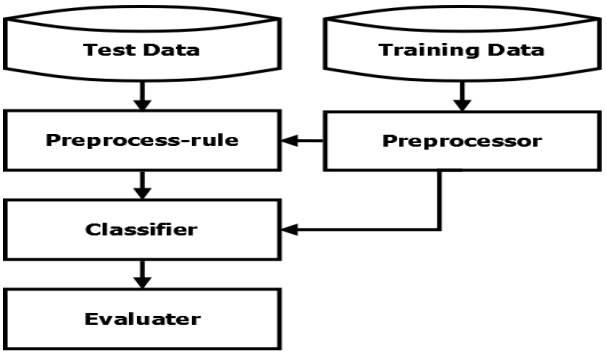
器結合[4, 17]。一些研究中顯示搭配多個分類器組合可以提升分類效能，網路入侵偵測的應用中，同樣加入預處理過程時多分類器方法較單一分類器方法有更高的分類精確度，但多分類器結合方式未有考慮資料不平衡與個別分類器獨立的預處理[3]。也有研究使用多分類器獨立預處理提升分類效能，將資料內各類經過不同屬性選擇後，由不同分類器分類出各類資料以增進分類精確度，但多分類器結合方式過於強調各類的分類精確度，使得一筆資料可能被分為多類[15]。或是利用多種不同分類演算法階層式結合，對於不同的類使用不同演算法加以偵測，分類結果雖然不會有重複分類情況發生，但訓練分類器使用的訓練集經過取樣以平衡資料集，卻沒有提供取樣比例的依據[27]。

不平衡資料集的處理分為兩方面有資料層面與演算法層面，資料層面的處理方式有過取樣(over sampling)、欠取樣(under sampling)或是合成取樣(synthetic sampling)等等，有些網路入侵偵測方式利用取樣方式平衡資料集以提升偵測精確度，但取樣方式與沒有依據[15, 25, 27]，演算法層面則有一些研究將增進方式如 Adaboost 等[16]，但由於網路入侵行為是不斷變化的，因此 boosting 方式用於不平衡的網路入侵偵測較容易發生過適(overfitting)情況。

本研究提出的階層式多分類器方法，為提升分類精確度使用適合網路入侵偵測的預處理方式，考慮資料內部不平衡問題與網路入侵得錯誤分類成本而依據類的大小決定分層順序，為提升多個分類器效果對個別分類器獨立預處理訓練資料集，未免資料被重複分類或不分類而採用階層式結合方法以免資料分類不清，基於保存資料集內部資訊以及原始資料分布因此不對資料集取樣。

## 參、分類器

基本的單一分類器架構。如**圖一**所示，對訓練資料集做預處理後產生預處理規則，測試資料用同樣的規則做預處理。利用訓練資料訓練分類器後，對測試資料做預測並評估效能。




圖一：基本分類器運作架構

在機器學習領域中，任何的訓練資料中可能包含雜訊或是附帶太多重複訊息，預處理可能可以提升演算法精確度與降低計算成本，但在不同資料集與不同演算法的環境，

預處理對分類精確度的影響是不一定能預測的[8]。預處理包含許多方式，如離散化數值屬性(如：EWD、EFD、EMD、PKID)、主成分分析(如：PCA、PIA)、屬性選擇(如：CFS、CONS、GN)以及預聚類資料(如：K-means)等等。使用分類器時常常結合多種預處理方式以提升分類效能，如結合離散化與屬性選擇方法找出適合入侵偵測系統的最佳預處理演算法組合[3]。

用於分類的資料集可能包含數種不同類型屬性，如整數、實數、類別等等，其中包含最多資訊的類型就是數值類型，但不同演算法適用的屬性類型不盡相同，多數分類演算法無法直接使用連續數值類型作為輸入，因此需要離散化產生新的類別型以便分類運算。好的離散化方式可以在保留較多資訊的同時，將屬性離散成數個適當的類別以方便計算，本研究離散化方式統一使用比例 K 區間離散法(Proportional k-interval discretization，簡稱 PKID) [29]離散各個資料集的屬性。

訓練資料集可能包含有許多無關分類或是冗餘的屬性，這些屬性往往影響最終的分類效果，而使用離散化後的屬性選擇可以有效的去除無關以及冗餘[18]，消除無關屬性可以有效降低雜訊提升判斷效率，而去除冗餘屬性則能使學習速度提升同時提供更易說明的屬性關聯性。資料集在經過屬性選擇後減少了屬性數量，達到節省資料儲存空間及降低運算成本的目的。

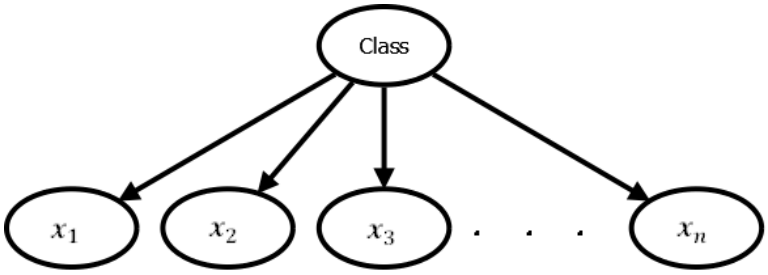
屬性選擇演算法有多種類型，由於本研究著重部分不在於屬性選擇方法之間的差異，且其他研究顯示在各離散化及屬性選擇中，基於一致性的屬性選擇(Consistency-based feature selection，簡稱 CONS)[9]對 KDD CUP 99 資料集表現優秀[3]，因此本研究選擇 CONS 作為屬性選擇演算法。

分類演算法為分類器的主體。由於網路入侵資料集實屬龐大，本研究由現今機器學習常用的分類演算法，如 SVM、類神經網路、貝氏機率和決策樹中選出訓練以及預測複雜度較低的貝氏演算法與決策樹演算法作為比較。

Bayesian Network 是依一組變數的相關性建立而成的機率圖模型[13, 28]。圖中的任一節點代表一個隨機變數，任一有向邊代表兩隨機變數間有因果相依性，最終建立完成必須是有向無循環圖，未連結的節點代表隨機變數之間視為相互獨立，而在父節點與子節點之間存在條件機率分布。

Naïve Bayes 演算法是 Bayesian Network 的一個特例。存在一個類節點為所有其他節點的根節點，即為期望推知的潛變量。其他節點代表觀察到的變量，觀察變量相互之間不存在邊，即除類節點以外所有隨機變數皆有條件的獨立[2]。Bayesian Network 架構如**圖二**，其中 Class 節點為將要被分類的潛變量。由於所有屬性節點視為獨立，因此基於 Naïve Bayes 演算法的聯合機率分布為：

$$\mathcal{P}(\mathbb{X}) = \mathcal{P}(Class = x_{class}) \prod_{i=1}^{n} \mathcal{P}(\mathrm{x}_i = x_i | Class = x_{class}) \quad (1)$$

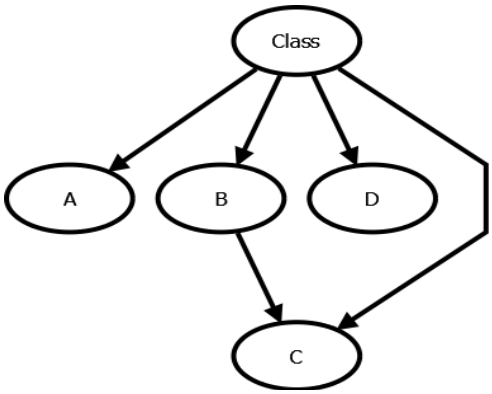



圖二：Naïve Bayes 分類器的 Bayesian Network 有向無循環圖示意

利用此機率模型建立的分類器就是 Naïve Bayes 分類器，雖然在現實空間中 Naïve Bayes 機率的獨立假設往往太強而不成立，但是其演算法已經可以提供一個快速且尚能接受的預測精確度。

Bayesian Network 分類器加入根節點以外的隨機變數之相依性，建立更為準確的機率圖模型。以**圖三**為例，其中 Class 節點為將要被分類的潛變量，其聯合機率分布為：

$$\mathcal{P}(\mathbb{X}) = \mathcal{P}(Class = x_{class})\mathcal{P}(A = x_a|Class = x_{class})$$
$$\mathcal{P}(B = x_b|Class = x_{class})\mathcal{P}(D = x_d|Class = x_{class})\mathcal{P}(C = x_c|B = x_b, Class = x_{class}) \quad (2)$$




圖三：Bayesian Network 分類器的一個簡單 Bayesian Network 有向無循環圖例子

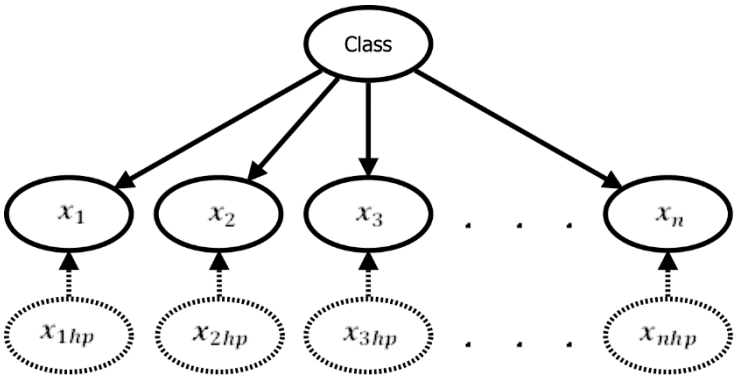
Bayesian Network 模型建立後即能夠有極佳的分類效果，但建立模型往往不容易。變數少量且意義明確時，可以由領域的專家確立各變數間的相依關係，但網路入侵偵測領域中資料量極大，參數很多且富有變化，利用人工方式建立模型費時費力，因此必須在自動方式搜尋網路結構同時評估結構的適用性。本研究重點並非在 Bayesian Network 架構搜尋方法的選擇，因此直接以爬山演算法(hill climbing algorithm)作為結構搜尋方式[7]，以簡單估計(simple estimate)作為結構評估方式。

Hidden Naïve Bayes(簡稱 HNB)分類演算法同屬貝氏機率類型，可用圖表示各屬性間相依性，如同 Naïve Bayes 演算法，各屬性為獨立的子節點同時連結身為父節點的類節點。以**圖四**為例，在 HNB 演算法中加入對各屬性間相依性做了補償運算，計算各屬性子節點的貝氏機率時需要採用各屬性間的相關程度加權值修正，如同屬性子節點各自

具有一個隱藏父節點(hidden parent)作為貝氏機率計算的修正[17]。其聯合機率分布計算如下：

$$\mathcal{P}(\mathbb{X}) = \mathcal{P}(Class = x_{class}) \prod_{i=1}^{n} \mathcal{P}\left(\mathbb{x}_i = x_i \middle| \mathbb{x}_{ihp} = x_{ihp}, Class = x_{class}\right) \quad (3)$$

其中 x_ihp 為 x_i 節點的隱藏父節點。




圖四：Hidden Naïve Bayes 網路圖示意

決策樹是用於決策論中輔助決策的工具，在機器學習之中也有非常重要的地位，可用來作為數據分析使用或是作為預測模型。作為預測模型時，由訓練資料集的內容決定決策樹的型式，從根節點開始選擇適當的屬性作為資料分割的依據，將資料集依所選擇的屬性分割到各個子節點，每個子節點再依照新選擇的適當屬性將子資料集再分割到下一層子節點。整個分割過程達到事先決定的指標即停止，如分類錯誤率降到一定數值或樹的高度達到閥值。

決策樹的實現有許多種，其中由 Ross Quinlan 提出的 C4.5 決策樹在機器學習以及資料探勘領域被廣為使用[19]。C4.5 決策樹演算法修改自 ID3 決策樹演算法，將屬性選擇基準由原本的資訊熵(information entropy)修改為資訊增益(information gain)及增加產生樹後剪枝部分。本研究中使用 J48 版本，是 WEKA 程式提供以 java 實現的 C4.5 決策樹演算法[25]。

## 肆、研究方法

介紹實驗方法，從測試資料集開始包含 KDD CUP 99 與 ND-KDD 以及其他蒐集自網路的資料，其次介紹實驗的階層式分類器架構，最後說明總體評估演算法效能的參數。

對於監督異常偵測的 IDS 的評估，需要標記的訓練資料集及標記的測試資料集，前者標記用於預測模型的訓練，後者標記用於學習成果的評估。因為入侵技術是日新月異的，異常偵測最好能對於新式攻擊有一定的偵測效率，所以測試集與訓練集的統計分佈必須不同，也就是說兩者不宜全部從同一資料集隨機選出。本文採用的資料集為 KDD CUP 99 資料集以及修改自前者的 ND-KDD 資料集。

KDD CUP 99 資料集源自 KDD CUP 資料探勘比賽，此比賽由世界計算機組織(Association for Computing Machinery，簡稱 ACM）轄下的資料探勘組(Special Interest Group on Knowledge Discovery and Data Mining，簡稱 SIGKDD)舉辦。自從 1997 年起至今每年舉辦一次，是資料探勘領域重要的比賽，而其中 99 年用於比賽的入侵偵測資料集不只包含訓練集與測試集，更有適當的評估方式以便比賽使用，因此 KDD CUP 99 資料集在入侵偵測領域被廣為使用。

資料分為五百萬筆的全部訓練集與約五十萬筆的百分之十訓練集以及三十萬筆的測試集，其中包含的攻擊數據可以分為四大類[1]，服務阻斷攻擊(Denial-of-Service，簡稱 DoS)、來自遠端的未授權存取(Remote to Local，簡稱 R2L)、未授權卻試圖取得超級管理員權限(User to Root，簡稱 U2R)、監控或其他探測(Probing，簡稱 Probe)。

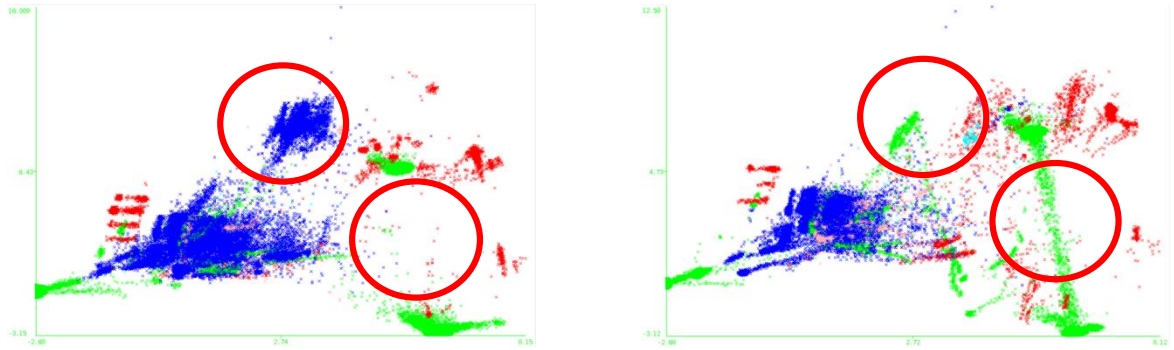
用於訓練與測試的數據往往從同一資料集內部隨機產生，因此訓練集與測試集的統計與空間分布相似，而 KDD CUP 99 資料集的訓練與測試集卻是兩個不相似的資料集。訓練資料集包含 24 種攻擊類型作為建立預測模型，測試資料集統計分布與訓練數據不同，另外還加入 17 種不同的攻擊，作為檢驗演算法是否能夠抓出新型態攻擊的評估。本研究取 10%訓練資料集作為訓練集，以全部的測試集做訓練結果的評估，資料集各類統計分布如**表一**，可以看出訓練與測試資料集的各類比例不相同，特別是 U2L 與 R2L 兩類數據所佔比例在測試資料集中較訓練資料增為 7 倍與 18.6 倍。

表一：KDD CUP 99 訓練與測試集各類統計分布

| KDD CUP 99 | Training data | Test Data |
|---|---|---|
| Total data number | 494021 | 311027 |
| Normal rate | 19.69% | 19.48% |
| Probe rate | 0.84% | 1.34% |
| Dos rate | 79.24% | 73.90% |
| U2L rate | 0.01% | 0.07% |
| R2L rate | 0.28% | 5.21% |

訓練與測試集資料之間的差異除了各大類所佔比例不同之外，還有在高維度空間中資料散布位置也有所不同。由於高維度空間的圖形無法直接呈現，因此需要運用一些降維工具才能觀察到這樣的現象，主成分分析(Principal Component Analysis，簡稱 PCA)就是一個廣為使用的成分分析、降維以及雜訊消除工具[26]。PCA 利用資料的分散狀況選取最佳分散方向以旋轉資料維度，在某些情況可以獲得較低的雜訊或較低的維度以增進分類器效能[20]。**圖五**展示了經過同樣的主成分分析旋轉方式後，投影在低階主成分的資料分布情況，一種顏色代表一大類的資料。藍色為正常連接；紅色為 Probe 類攻擊；綠色為 DoS 類攻擊；淺藍色為 U2R 類攻擊；淺紅色為 R2L 類攻擊。由兩圖中的紅圈可

以看出有部分的測試資料在空間中的分布與訓練資料不同，這樣的不同分布是精確分類資料的困難。



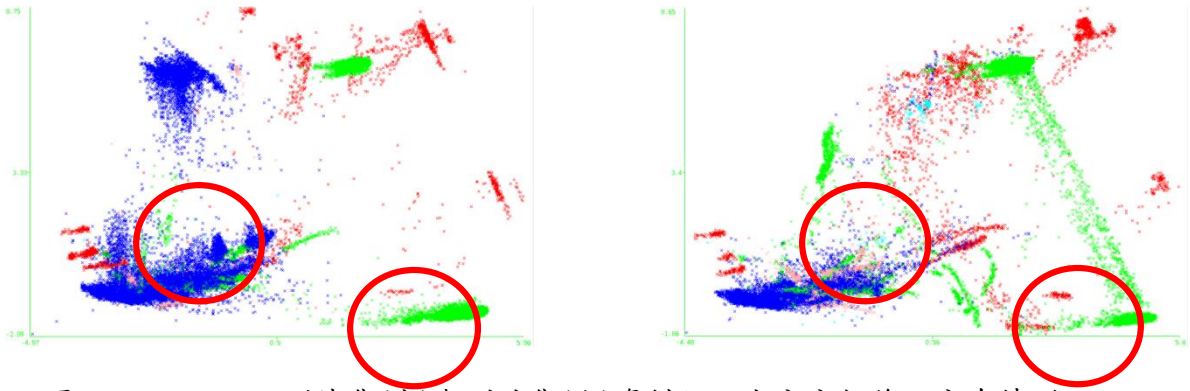
圖五：KDD CUP 99 訓練集(左)與測試集(右)資料經主成分分析後之分布情形

KDD CUP 99 資料集中存在著數量可觀的重複數據，以統計模型作為預測的演算法會受到重複數據多寡而影響權重。因此將 KDD CUP 99 資料集的五類數據中所有重複的資料刪除只保留一筆，作為新的非重複資料集(Non-Duplicate KDD, 簡稱 ND-KDD)測試，也有其他研究使用同樣概念[22, 24]。藉由**表二**同樣可以看出，在去除重複資料後訓練及測試資料集的統計分布依然不同，除此之外去除重複資料的資料集內部五類資料數量比例與表一相比差距減小，且最多筆資料的類從 DoS 換成 Normal 連接。

表二：ND-KDD 訓練與測試集各類統計分布

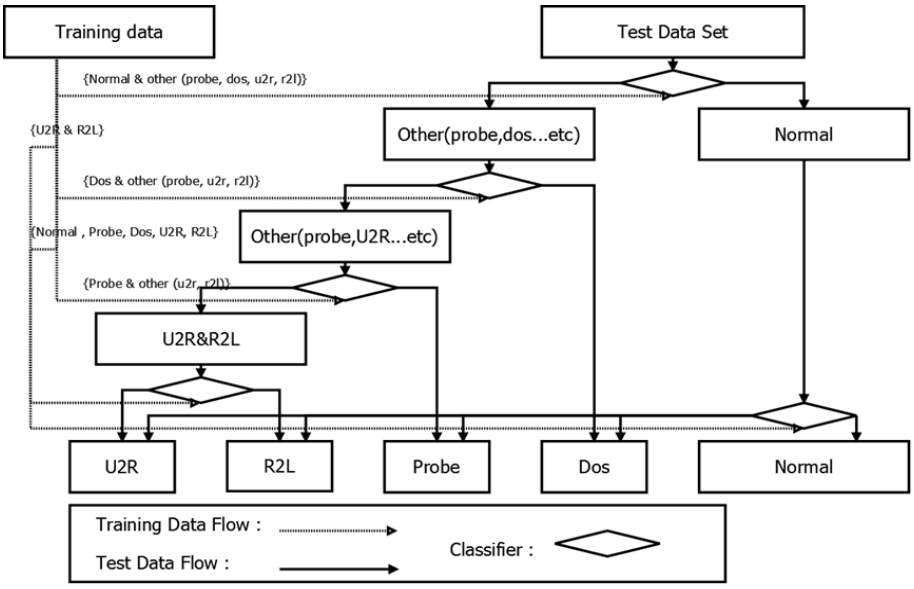
| NL - CUP 99 | Training data | Test Data |
|---|---|---|
| Total data number | 145585 | 77285 |
| Normal rate | 60.33% | 61.99% |
| Probe rate | 1.46% | 3.47% |
| Dos rate | 37.48% | 30.49% |
| U2L rate | 0.04% | 0.28% |
| R2L rate | 0.69% | 3.77% |

經主成分分析觀察，**圖六**紅圈內資料點顯示 ND-KDD 去除重複資料後訓練與測試資料集空間中分布依然不同，代表經過處理的資料集還是很適合作為分類器的評估資料集。



圖六：ND-KDD 訓練集(左)與測試集(右)資料經主成分分析後之分布情形

在入侵偵測領域中，各種攻擊分佈相較於正常往往是佔少數的，各種攻擊間的比率也相差懸殊，為了提升對攻擊的捕捉效率將較少數的攻擊從較多數的攻擊中分離出來，而提出了階層式的多分類器分類架構。此架構利用組合多個二元分類器與多類分類器以提升對數量較少攻擊的分類精確度。**圖七**說明用於 KDD CUP 99 以及 ND-KDD 資料集的分類器架構圖，虛線箭頭代表提供各分類器的訓練資料，實線箭頭代表測試資料分類前後的流向。




圖七：階層式多分類器架構

在機器學習領域中，不平衡的資料集會造成偏頗的分類，分類器會偏重於偵測較大的類而忽略較小的類，然而，在某些現實情境中找出屬於較小類的資料比正確分出最多資料還要重要。網路入侵偵測領域大多數情況符合前文所述，偵測佔小比例的攻擊往往比大宗的正常資料還要來得重要，因此需要針對一些不平衡網路入侵資料集重新平衡修正，以提升小類攻擊的偵測率，但有時最大類可能不是正常類，因此分層方式必須兼顧兩種情況。

本研究在多二元分類器架構上，提出一個結合少數類同時兼顧 IDS 特性的方式重新平衡資料集，分層方式需要兼顧兩種個方面，一是錯誤成本差異，二是小類資料偵測。由於類合併後被偵測率較高相當於偵測權重較高，因此考慮兩種分層原則，第一種是成本優先，正常連接自成一類，攻擊類合併以包容多類攻擊特徵增加預測多樣性，提升對新攻擊的偵測率；第二種是類大小優先，較小類合併以平衡資料中極端小的類，提升只佔少數的類的偵測率。

經實驗表明使用 KDD CUP 99 資料集時，成本考量先於類大小，而 ND-KDD 資料集成本與類大小考量一致，實驗以兩原則依資料量由多至少分階。由於資料集共分五類，

使用四個階層的二元分類器加上考量成本因素的第五分類器共五階:

(1) 一階訓練集：由於 IDS 較不宜將攻擊分類為正常，因此將正常標記為一類且其他視為一類，此二元訓練集有助於偵測出較多攻擊。此二元分類資料經預處理後，得到一組用於訓練與測試的屬性組合[3]。

(2) 二階訓練集：訓練集去除正常連接資料後，標記 DoS 資料為一大類，其他三類攻擊資料為一大類。此二元訓練集經預處理後得到一組專屬的屬性組合。

(3) 三階訓練集：訓練集去除正常與 DoS 資料後，標記 Probe 資料為一大類，其他兩類攻擊為一大類，此二元訓練集經預處理後得到一組專屬的屬性組合。

(4) 四階訓練集：由 U2L 類與 R2L 類組成，同樣經預處理後得到適合的屬性子集。

(5) 五階訓練集：使用原始標記為五類的資料作為多類分類器的訓練資料，經預處理後得訓練用屬性子集。

二至四階分類器劃分訓練集規則依各類大小相互結合，最大量的類單獨為一類其餘合併為一類。最後的第五階分類器視為對分類的最後補充，將階層分類後部分被預測為正常的資料分類到各別攻擊類中。

訓練資料集先經過類合併產生適合各分類器的子訓練集，接下來對各訓練集做預處理及產生預處理規則，預處理全部採用 PKID 演算法以及由 CONS 評估的屬性選擇演算法。為了統一資料屬性內容以及訓練的公正性，測試集不參與預處理方式的建立，而是在進入分類器之前依照訓練集產生的預處理規則做處理。個別化的預處理有許多好處，除了離散化的結果以及選擇出的屬性能夠更貼近個別分類，同時降低因資料類別多時選擇過多屬性而產生的過適現象。使用 KDD CUP 99 與 ND-KDD 資料集的各階層屬性選擇結果見**表三**。

表三：KDD CUP 99 屬性選擇

| KDD CUP 99 | Selected features |
|---|---|
| Level 1 | 1，3，5，32，34，39 |
| Level 2 | 2，4，5，8，9，22，23，24，27，35，38，40 |
| Level 3 | 1，2，4，5，26，35 |
| Level 4 | 1，3，6，7，15，19，20，34，37 |
| Level 5 | 3，4，5，11，12，13，15，21，23，25，26，29，32，33，35，38，41 |

將由評估函數比較實驗結果，本研究從三方面評估效果，首先是分類器的精確度與錯誤率，其次是在 KDD CUP 99 比賽中基於分類錯誤成本的平均成本計算方式，最後是針對未知攻擊的偵測比率。

當分類器所需分類的類總數共有 K 類時，分類結果由一個 K*K 的方陣呈現。此方陣稱為混淆矩陣（confusion matrix），方陣第 i 列第 j 行放置的數字代表測試集中標示為 i 類卻分類為 j 類的資料個數。評估分類器效能可以使用精確度與錯誤率，兩者相

加為 1，精確度(Accuracy)越高越好、錯誤率(Error Rate)越低越好，其計算函式如下所示：

$$\text{Accuracy} = \frac{1}{N}\sum_{i=1}^{K}\mathcal{M}(i,i) \quad (4)$$

$$\text{Error Rate} = 1 - \text{Accuracy} \quad (5)$$

其中 N 為測試集資料總數，K 為類別總數，M(i,i)為混淆矩陣第 i 列第 i 行的值。

在 KDD CUP 99 比賽中的評估函數並非使用精確度函數，而是加入一個考量網路入侵偵測特性的權重方陣。如**表四**方陣，此方陣稱為成本矩陣（cost matrix），第 i 列第 j 行放置的數字代表入侵偵測中，第 i 類卻被分類為第 j 類的資料時產生的分類成本，由於各種錯誤分類會產生不同的後果，因此必須針對性的給予不同的權重。分類錯誤結果對系統安全性影響越大者，應當在評估過程中給予較大的分類錯誤成本，如：正常連接被分類為 R2L 與 DoS 攻擊被分類為 R2L 攻擊兩種，同樣是分類錯誤，但前者成本權重為 4 後者為 2，也就是說前項錯誤影響系統安全性較後者來得大，若是使用其他資料集，也需要依此原則制定錯誤成本。平均成本(Average Cost)是越低越好，而計算函數如下：

$$\text{Average Cost} = \frac{1}{N}\sum_{i=1}^{K}\sum_{j=1}^{K}\left(\mathcal{M}(i,j) * \mathcal{C}(i,j)\right) \quad (6)$$

其中 N 為測試集資料總數，K 為類別總數，M(i,j)為混淆矩陣第 i 列第 j 行的值，C(i,j)為成本矩陣第 i 列第 j 行的權重。

表四：用於 KDD CUP 99 資料集評估的成本矩陣

| | Normal | Probe | Dos | U2R | R2L |
|---|---|---|---|---|---|
| **Normal** | 0 | 1 | 2 | 2 | 2 |
| **Probe** | 1 | 0 | 2 | 2 | 2 |
| **Dos** | 2 | 1 | 0 | 2 | 2 |
| **U2R** | 3 | 2 | 2 | 0 | 2 |
| **R2L** | 4 | 2 | 2 | 2 | 0 |

在 KDD 資料集中，測試集多出 17 種在訓練集中沒有的攻擊以模擬偵測新型態攻擊。異常偵測對新型態攻擊應當有效果，因此作為監督型分類器對新式攻擊的偵測分析，將以表格的方式條列比較 17 種新攻擊的偵測率。

## 肆、實驗結果

比較階層式多分類器與單一多類分類器及其他類似結合多分類器結果。演算法部分使用 WEKA 及部分 C++語言撰寫的程式，WEKA 是一款由 java 編寫而成的受歡迎免費工具軟體，符合 GNU 通用公共授權條款，用於機器學習以及資料探勘演算法的研究，除了軟體介面本身還另外提供 API 可以在自己的程式中使用。實驗環境的作業系統為 Ubuntu 14.04，硬體部分為 Intel(R) Xeon(R) CPU E5-1620 0 @ 3.60GHz、64GB RAM。由於分層方式可以有許多不同且 KDD CUP 99 資料集分類錯誤成本與類大小不一致，因

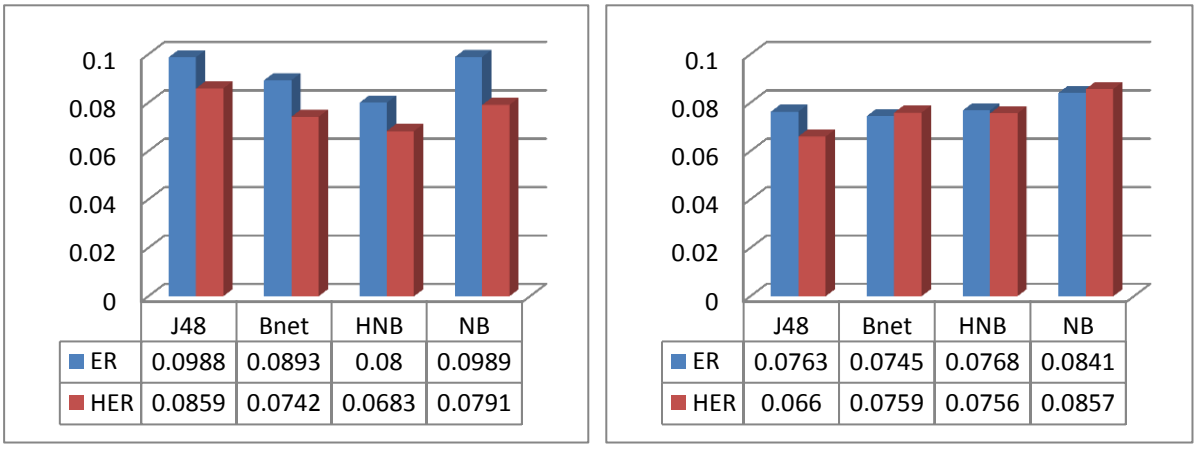
此將比較成本優先考量與類大小優先考量兩種方式，成本考量方式為首先將資料中攻擊類與正常類分開而後才依照類大小分層，類大小考量方式完全依照類大小將資料分層。

**表五**為 KDD CUP 99 資料集使用兩種分層方式的差異，演算法使用 C4.5 決策樹，數字顯示加入成本考量的部分較單純類大小考量來得優秀。前者除了類大小外加入額外成本考量，可以預期的平均成本較單純類大小考量來得低，但讓人訝異的是錯誤率也較單純類大小來得低，可能原因是 KDD CUP 99 資料集的各種類攻擊間相似，因此考慮錯誤成本的分層方式，第一步合併所有攻擊類恰好符合資料集內部的趨勢，因此以下實驗都使用成本考量的分層方式。

表五：兩種分層方式結果差異

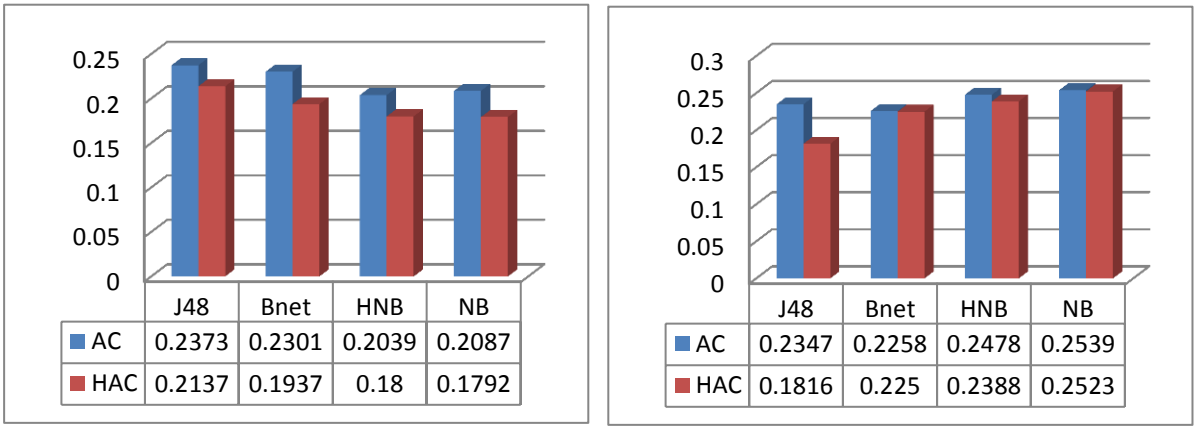
| | 成本考量 | 類大小考量 |
|---|---|---|
| Error Rate | 0.066 | 0.0744 |
| Average Cost | 0.1816 | 0.2274 |

本研究實驗結果將比較提出的階層式方法與單一的多類分類器的錯誤率、平均成本以及對於新攻擊的偵測率，演算法選用 C4.5 決策樹、Naïve Bayes 分類器、Bayesian Network 分類器及 HNB 分類器，資料集使用 KDD CUP 99 資料集與 ND-KDD 資料集。剔除副本的 ND-KDD 資料集實驗中，階層式方法在搭配各分類演算法後都有很好的表現，見**圖八**中藍色代表單一多類分類器的錯誤率(ER)，紅色是階層式多分類器的錯誤率(HER)。總體而言本研究方法相較於單一多類分類器，降低錯誤率平均達百分之十六，其中貝氏系列演算法較決策樹 C4.5 有更低的錯誤率。而**圖八**的 KDD CUP 99 資料集實驗結果與 ND-KDD 實驗不同，使用 KDD CUP 99 資料集測試時 C4.5 決策樹表現較貝氏系列演算法佳。顯示提出的階層式方法在 KDD CUP 99 資料集搭配貝氏系列演算法的精確度提升效果不顯著，特別是在 Bayesian Network 分類器與 Naïve Bayes 分類器都造成錯誤率略微上升。




圖八：ND-KDD(左)與 KDD CUP 99(右)比較單一多類分類器及階層式多分類器錯誤率

在 ND-KDD 資料集測試中平均成本的結果與錯誤率表現相同，如**圖九**中的藍色代表單一多類分類器的平均成本(Average Cost，簡稱 AC)，紅色是階層式多分類器的平均成本(Hierarchical Average Cost，簡稱 HAC)。圖中各分類演算法在加入階層式方法修正後平均成本都下降，顯示階層式多分類器方法對平均成本有穩定的下降效果，四種演算法中平均成本平均降低百分之十三，由於網路異常偵測中預測資料的數量級非常的大，因此相對於單一多類分類器下降了百分之十三的平均成本，在考慮實際流量時是有其價值的。而加入階層式方法對 KDD CUP 99 資料集的測試結果，即使 Bayesian Network 與 Naïve Bayes 分類器的錯誤率部分略為提升，但是所有演算法的平均成本都還是下降，其中 C4.5 決策樹演算法的表現優於貝氏系列演算法。




圖九：ND-KDD(左)與 KDD CUP 99(右)比較單一多類及階層式多分類器平均成本

由於異常偵測相較於誤用偵測的優點就在於對新型態的攻擊有效果，因此試著挑出測試集內的新型態攻擊比較差異。見**表六**，ND-KDD 資料集的實驗結果表示四大類的所有新型態攻擊偵測率都有增加，被錯誤分類的正常連接只有微量增加，推測是因為 ND-KDD 資料集正常類為最大宗，四類攻擊偵測都因分層而受益。而 KDD CUP 99 資料集結果表示除了 DoS 攻擊以外，其他三類屬於小量的新型態攻擊預測率都有明顯提升，推測是由於 DoS 類原本即為最大宗攻擊類，因而使此類預測率只有些微提升。

表六：ND-KDD 與 KDD CUP 99 資料集的新攻擊偵測

| | | *Total ND-KDD* | *Single* | *Hierarchical* | *Total KDD CUP 99* | *Single* | *Hierarchical* |
|---|---|---|---|---|---|---|---|
| **Dos** | Apache2 | 794 | 400 | 493 | 794 | 747 | 747 |
| | mailbomb | 308 | 0 | 0 | 5000 | 0 | 0 |
| | processtable | 744 | 551 | 612 | 759 | 511 | 759 |
| | udpstorm | 2 | 2 | 2 | 2 | 0 | 2 |
| **Probe** | mscan | 1049 | 125 | 273 | 1053 | 69 | 765 |
| | saint | 364 | 349 | 352 | 736 | 714 | 723 |
| **R2L** | httptunnel | 145 | 1 | 1 | 158 | 0 | 1 |
| | named | 17 | 1 | 2 | 17 | 0 | 3 |
| | sendmail | 15 | 0 | 3 | 17 | 0 | 2 |
| | snmpgetattack | 179 | 0 | 0 | 7741 | 0 | 1 |
| | snmpguess | 359 | 0 | 0 | 2406 | 0 | 2403 |
| | worm | 2 | 0 | 0 | 2 | 0 | 2 |
| | xlock | 9 | 1 | 2 | 9 | 0 | 2 |
| | xsnoop | 4 | 1 | 2 | 4 | 0 | 0 |
| **U2R** | ps | 16 | 2 | 3 | 16 | 0 | 9 |
| | sqlattack | 2 | 0 | 0 | 2 | 0 | 0 |
| | xterm | 13 | 3 | 4 | 13 | 0 | 4 |

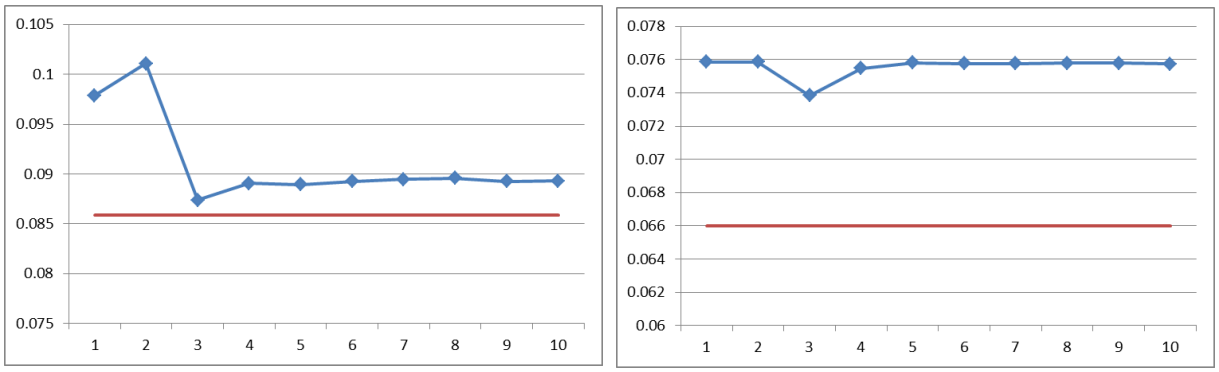
在 KDD CUP 99 比賽中，決定績效的就是平均成本，因此本研究也與一些其他多類分類研究比較平均成本。**表七**中，除了探討單一分類器效果[11, 17, 30]，也有一些探討多分類器擬合的結果[3, 21]，其中階層式多分類器搭配 C4.5 決策樹演算法的平均成本，在其他不加入取樣方法的演算法中是最低的，平均成本較 KDD CUP 99 獲勝隊伍降低了百分之二十二點一。

表七：平均成本與其他研究的比較

| Approach | Average Cost |
|---|---|
| **Winner of KDD CUP 99[11]** | 0.2331 |
| **Misuse Detection Context[21]** | 0.2285 |
| **Hidden Naïve Bayes multiclass classifier[17]** | 0.2224 |
| **Feature selection and classification[3]** | 0.2123 |
| **Parzen-window network intrusion detectors[30]** | 0.2024 |
| **Hierarchical Multi-classifier** | 0.1816 |

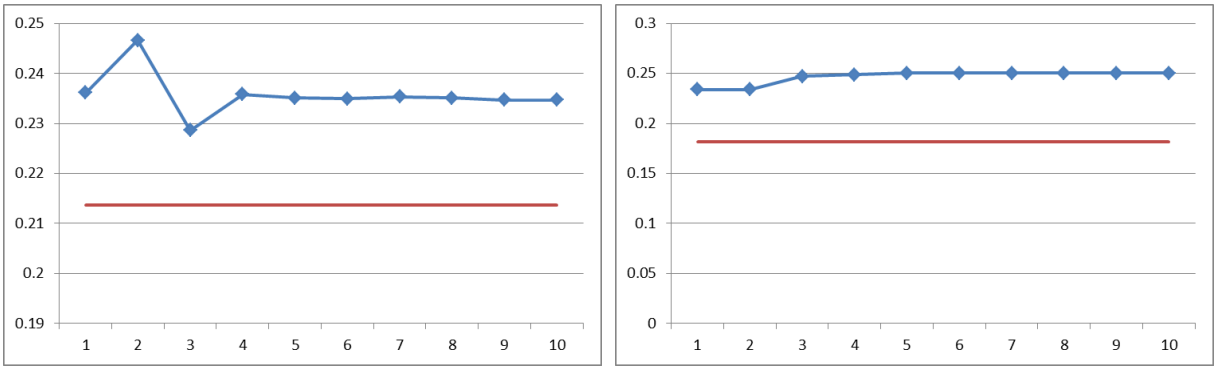
對於網路入侵不平衡資料集有其他如 Adaboost 方法[16]，使用多個弱分類器提升偵測精確度，可對不容易分類出的小類資料重新分類。Adaboost 方法是演算法層面的增進

方式，在 weka 工具上有實作。**圖十**是使用 Adaboost 方法搭配 C4.5 決策樹演算法結果，縱軸是錯誤率、橫軸是 Adaboost 方法的迭代次數、紅色線是同樣資料集使用階層式方法搭配 C4.5 決策樹演算法的實驗結果。可以看到 ND-KDD 資料集方面 Adaboost 方法在迭代三次時表現接近階層式方法，但沒有勝過階層式方法，迭代四次以上時可能因為過適反而錯誤率上升，KDD CUP 99 資料集部分同樣是三次迭代未達階層式方法效果且迭代次數上升效果差。




圖十：ND-KDD(左)與 KDD CUP 99(右)資料集使用 Adaboost 方法錯誤率

**圖十一**顯示 Adaboost 方法搭配 C4.5 決策樹演算法結果，縱軸是平均成本、橫軸是 Adaboost 方法的迭代次數、紅色線是同樣資料集使用階層式方法搭配 C4.5 決策樹演算法的實驗結果。與錯誤率相同的是，雖然部分迭代次數下平均成本有所下降，但皆未達到階層式方法部分，是因為階層式方法有特別考慮錯誤成本且網路入侵偵測資料是動態的時時刻刻會產生新的類型的攻擊資料，Adaboost 方法容易因迭代次數過多而預測失準。




圖十一：ND-KDD(左)與 KDD CUP 99(右)資料集使用 Adaboost 方法平均成本

除了用於 IDS 的 KDD CUP 99 資料集之外，階層式多分類器方法也可以對其他的不平衡資料集作出修正預測，本研究加入了一些網路入侵偵測以外的不平衡資料集測試，檢驗對於非網路入侵資料集的效果。KEEL 提供了一些有關於不平衡資料集的檔案，取其中一些適用的資料集略做修改以配合檢驗[10]。

因為資料集的資料有數量與屬性的差異，因此挑出三個適用的資料集並對資料集內容及實驗方式略做修正，由於使用的三個資料集的資料量不大，因此實驗統一採用三次交叉驗證，訓練集與測試集隨機挑選，且屬性數目不多將不進行屬性選擇的步驟。採用的分類演算法為在 KDD CUP 99 資料集中表現最好的 C4.5 決策樹演算法。使用的資料集如下：

1. 酵母菌(yeast)

   資料集來自真實世界，利用觀察到的特性將酵母菌分類。原始資料集共分為 10 類，8 個屬性，1484 筆資料。挑選其中 5 類共 727 筆資料，形成不平衡的子資料集作為使用的資料集。5 類分別是 CYT、ME3、ME2、VAC 以及 POX 類別。

2. 玻璃(glass)

   資料集來自真實世界，檢測玻璃組成元素種類進而分類。原始資料集共分 7 類，9 個屬性，214 筆資料。挑選其中 5 類共 138 筆資料，形成不平衡的資料集。5 類分別是 1、7、3、5 以及 6。

3. 頁面塊(pageblocks)

   資料集來自真實世界。原始資料集共分 5 類，10 個屬性，548 筆資料。五類為 1、2、4、5 以及 3。

**表八**比較階層式多分類器以及單一多類分類器對各類分類精確度，類排序由左至右為最大到最小，單位皆為百分比。其中酵母菌測試集顯示階層式方法提升總精確度，第三類類偵測精確度也從 0%提升至 49%；玻璃測試集數據顯示，最大數據量類之外的其他小類偵測率與總精確度皆提升；頁面塊測試集的總精確度雖然呈現持平狀態，但類別 3、4 小類都有所提升；總而言之，階層式多分類器可以達到對相對較小數量的類偵測率提升，同時不損及總體偵測精確度。

表八：其他資料用於階層式多分類器比較

| | | Class 1 | Class 2 | Class 3 | Class 4 | Class 5 | All |
|---|---|---|---|---|---|---|---|
| **Yeast** | ACC | 97.8 | 90.1 | 0 | 0 | 0 | 82.6 |
| | HACC | 96.3 | 90.1 | **49** | 0 | 0 | **85.1** |
| **Glass** | ACC | 94.3 | 82.8 | 5.9 | 38.5 | 22.2 | 71.0 |
| | HACC | 90.0 | **86.2** | **23.5** | **76.9** | **88.9** | **79.7** |
| **Pageblocks** | ACC | 98.4 | 36.4 | 25.0 | 0 | 33.3 | 91.2 |
| | HACC | 98.0 | 36.4 | **33.3** | **12.5** | 33.3 | 91.2 |

同樣的其他不平衡資料集可以使用 Adaboost 方法增進，**表九**列出使用 Adaboost 方式對其他不平衡資料集的十次迭代增進結果，Adaboost 方式在頁面塊資料集表現最佳，在第二類、第四類、第五類與總合來說精確度的提昇都超過階層式方法；在玻璃資料集 Adaboost 方法只有第三類精確度曇花一現般的高過階層式方法，其他部分精確度都較階

層式方法來得低；在酵母菌資料集 Adaboost 方法在總精確度方面雖然沒有勝過階層式方法，但最小的第五類與第四類精確度從0%提升到15.8%與3.3%較階層式方法來得好。三個資料集實驗中 Adaboost 方法與階層式方法結果各勝擅場，不似使用 KDD CUP 99 資料集時階層式方法壓倒性勝利，可能是因為 KDD CUP 99 資料集訓練集與測試集分布不同，導致過多的迭代次數後發生過適，而其他資料集實驗兩資料集分布類似，因此 Adaboost 表現提升。

表九：其他不平衡資料集使用 Adaboost 增進方式精確度

| | iterator | Class 1 | Class 2 | Class 3 | Class 4 | Class 5 | All |
|---|---|---|---|---|---|---|---|
| **Yeast** | 1 | 98.1 | 90.2 | 0 | 0 | 0 | 82.8 |
| | 2 | 94.8 | 82.2 | 11.8 | **3.3** | **15.8** | 80.3 |
| | 3 | 94 | 69.9 | 33.3 | 3.3 | 15.8 | 78.5 |
| | 4 | 86.6 | 73 | 35.3 | 3.3 | 15.8 | 74.7 |
| | 5 | 87.7 | 76.1 | 29.4 | 0 | 10.5 | 75.3 |
| | 6 | 91.1 | 77.3 | 35.3 | 3.3 | 5.3 | 78.2 |
| | 7 | 92.7 | 82.2 | 37.3 | 3.3 | 5.3 | 80.4 |
| **Glass** | 1 | 95.7 | 82.8 | 0 | 23.1 | 44.4 | 71 |
| | 2 | 87.1 | 75.9 | 17.6 | 46.2 | 22.2 | 68.1 |
| | 3 | 97.1 | 75.9 | 11.8 | 38.5 | 55.6 | 73.9 |
| | 4 | 88.6 | 75.9 | 5.9 | 38.5 | 55.6 | 68.8 |
| | 5 | 85.7 | 75.9 | 23.5 | 46.2 | 55.6 | 70.3 |
| | 6 | 88.6 | 79.3 | 23.5 | 53.8 | 44.4 | 72.5 |
| | 7 | 81.4 | 75.9 | **35.3** | 69.2 | 55.6 | 71.7 |
| **Pageblocks** | 1 | 99.2 | 27.3 | 8.3 | 0 | 33.3 | 91.1 |
| | 2 | 96.5 | **87.9** | 25 | 0 | **66.7** | 92.9 |
| | 3 | 99.2 | 60.6 | 16.7 | **37.5** | 66.7 | 94 |
| | 4 | 98.8 | 66.7 | 16.7 | 12.5 | 66.7 | 93.6 |
| | 5 | 98.2 | 84.8 | 25 | 25 | 66.7 | 94.5 |
| | 6 | 98.8 | 75.8 | 25 | 12.5 | 66.7 | 94.3 |
| | 7 | 98.2 | 87.9 | 25 | 25 | 66.7 | **94.7** |

在實際網路中攻擊與異常的存在相較於正常連結都是少量的且富變化的，但 KDD CUP 99 訓練集與測試集皆包含大量副本資料，導致評估各演算法時可能產生偏頗情形[24, 28]，主要影響有二者：

1. 分類器對於訓練集內部副本較多的類或資料有較高預測權重，即分類器偏好將資料分類為此類。
2. 評估函數將給予成功分類副本資料的分類器較高評價，即相對於分類無副本資料，正確分類有副本資料能夠得較高分。

在 KDD CUP 99 資料集的錯誤率實驗結果顯示，貝氏系列演算法應用階層式方法在 KDD CUP 99 資料集表現不佳，推測是貝氏系列機率演算法容易被統計分布以及密度影響，因此 KDD CUP 99 資料集含大量副本影響評估結果，同時分層兩原則在 KDD CUP 99 資料集中互相牴觸，使得階層式方法無法達到最好效果。平均成本部分雖然錯誤率有部分演算法略微上升，但由於分類錯誤成本有其權重不同，因此所有的演算法的平均成本依然皆為下降，而 C4.5 決策樹特別在 KDD CUP 99 資料集表現優秀，可能源於決策樹建構時較不易受大量副本影響，而在平衡後的分層架構增進下，階層式分類 C4.5 決策樹平均成本降低百分之二十三。

藉由上述可知，在較符合實際數據情況的 ND-KDD 資料集實驗中，階層式多分類器應用四種演算法都有明顯的增進效果，錯誤率及平均成本都有很好的降低。

## 肆、結論與未來方向

本研究提出一個基於階層式架構的多分類器擬合方式，可降低網路入侵偵測領域中因為不平衡資料集產生的分類誤差及平均成本，該方式利用個別的預處理方式及平衡式的分層原則，產生多個不同面向但使用相同演算法的分類器並擬合所有預測結果。

實驗結果顯示，階層式方法對於隱含不平衡類的網路異常偵測分類效果有所提升，在 ND-KDD 資料集結果中分類錯誤率平均降低百分之十六，平均成本平均降低百分之十三；在異常偵測的主要優勢「偵測新型態攻擊」方面，佔極小比例的 R2L 與 U2R 類新型態攻擊都偵測到更多；在其他不平衡資料集部分總分類精確度最高提升有百分之八，較小類的分類精確度也都提升有百分之四到百分之六十不等。對應日益增加的大量網路流量，降低分類成本有助於更快速的整理與發現新出現的異常，在新型態攻擊發生的初期即被發現且修復漏洞縮短零日攻擊的時間。階層式多分類器在演算法的選擇是富有彈性的，對於不同分類演算法，階層式多分類器架構也都有助於降低錯誤率以及平均成本。

平衡資料集也有其他方式，如過取樣、欠取樣或是合成取樣等等。在日後的工作中，預處理方面可以試著加入適當的取樣，使得取樣以及階層式平衡方式能相互結合，可能能夠針對少數類再提升偵測率及降低處理資料量以提升計算速度。未來對於演算法部分，可加入半監督或是無監督偵測提升對於新型態攻擊的偵測。

## 參考文獻

[1] ACMSIGKDD, "http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html". 1999.

[2] N. B. AMOR, S. BENFERHAT, and Z. ELOUEDI. "Naive bayes vs decision trees in intrusion detection systems". *in Proceedings of the 2004 ACM symposium on Applied computing*. 2004. ACM.

[3] V. BOLÓN-CANEDO, N. SÁNCHEZ-MAROÑO, and A. ALONSO-BETANZOS, "Feature selection and classification in multiple class datasets: An application to KDD Cup 99 dataset". *Expert Systems with Applications*, 2011. **38**(5): p. 5947-5957.

[4] V. BOLON-CANEDO, N. SANCHEZ-MAROO, and A. ALONSO-BETANZOS. "A combination of discretization and filter methods for improving classification performance in KDD Cup 99 dataset". *in Neural Networks, 2009. IJCNN 2009. International Joint Conference on*. 2009. IEEE.

[5] BRO. 2014; Available from: https://www.bro.org/index.html.

[6] V. CHANDOLA, A. BANERJEE, and V. KUMAR, "Anomaly detection: A survey". *ACM Computing Surveys (CSUR)*, 2009. **41**(3): p. 15.

[7] G. F. COOPER and E. HERSKOVITS, "A Bayesian method for the induction of probabilistic networks from data". *Machine learning*, 1992. **9**(4): p. 309-347.

[8] S. F. CRONE, S. LESSMANN, and R. STAHLBOCK, "The impact of preprocessing on data mining: An evaluation of classifier sensitivity in direct marketing". *European Journal of Operational Research*, 2006. **173**(3): p. 781-800.

[9] M. DASH and H. LIU, "Consistency-based search in feature selection". *Artificial Intelligence*, 2003. **151**(1-2): p. 155-176.

[10] J. DERRAC, S. GARCIA, L. SANCHEZ, and F. HERRERA, "KEEL Data-Mining Software Tool: Data Set Repository, Integration of Algorithms and Experimental Analysis Framework". *Journal of Multiple-Valued Logic and Soft Computing* 2011.

[11] C. ELKAN, "Results of the KDD'99 Classifier Learning Contest". 1999.

[12] H. HE and E. A. GARCIA, "Learning from imbalanced data". *Knowledge and Data Engineering, IEEE Transactions on*, 2009. **21**(9): p. 1263-1284.

[13] D. HECKERMAN, *A tutorial on learning with Bayesian networks*, in *Innovations in Bayesian Networks*. 2008, Springer. p. 33-82.

[14] P. HELMAN, G. LIEPINS, and W. RICHARDS. "Foundations of intrusion detection [computer security]". *in Computer Security Foundations Workshop V, 1992. Proceedings.* 1992. IEEE.

[15] S.-J. HORNG, M.-Y. SU, Y.-H. CHEN, T.-W. KAO, et al., "A novel intrusion detection system based on hierarchical clustering and support vector machines". *Expert Systems with Applications*, 2011. **38**(1): p. 306-313.

[16] W. HU, W. HU, and S. MAYBANK, "Adaboost-based algorithm for network intrusion detection". *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, 2008. **38**(2): p. 577-583.

[17] L. KOC, T. A. MAZZUCHI, and S. SARKANI, "A network intrusion detection system based on a Hidden Naïve Bayes multiclass classifier". *Expert Systems with Applications*, 2012. **39**(18): p. 13492-13500.

[18] H. LIU and R. SETIONO, "Feature selection via discretization". *IEEE Transactions on knowledge and Data Engineering*, 1997. **9**(4): p. 642-645.

[19] J. R. QUINLAN, *C4. 5: programs for machine learning*. Vol. 1. 1993: Morgan kaufmann.

[20] J. J. RODRIGUEZ, L. I. KUNCHEVA, and C. J. ALONSO, "Rotation forest: A new classifier ensemble method". *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 2006. **28**(10): p. 1619-1630.

[21] M. SABHNANI and G. SERPEN. "Application of Machine Learning Algorithms to KDD Intrusion Detection Dataset within Misuse Detection Context". *in MLMTA*. 2003.

[22] S. S. SIVATHA SINDHU, S. GEETHA, and A. KANNAN, "Decision tree based light weight intrusion detection using a wrapper approach". *Expert Systems with Applications*, 2012. **39**(1): p. 129-141.

[23] SNORT. 2014; Available from: https://www.snort.org/.

[24] M. TAVALLAEE, E. BAGHERI, W. LU, and A.-A. GHORBANI. "A detailed analysis of the KDD CUP 99 data set". *in Proceedings of the Second IEEE Symposium on Computational Intelligence for Security and Defence Applications 2009*. 2009.

[25] WEKA. 2014; Available from: http://www.cs.waikato.ac.nz/~ml/weka/.

[26] S. WOLD, K. ESBENSEN, and P. GELADI, "Principal component analysis". *Chemometrics and intelligent laboratory systems*, 1987. **2**(1): p. 37-52.

[27] C. XIANG, P. C. YONG, and L. S. MENG, "Design of multiple-level hybrid classifier for intrusion detection system using Bayesian clustering and decision trees". *Pattern Recognition Letters*, 2008. **29**(7): p. 918-924.

[28] L. XIAO, Y. CHEN, and C. K. CHANG. "Bayesian Model Averaging of Bayesian Network Classifiers for Intrusion Detection". *in Computer Software and Applications Conference Workshops (COMPSACW), 2014 IEEE 38th International*. 2014. IEEE.

[29] Y. YANG and G. I. WEBB, *Proportional k-interval discretization for naive-Bayes classifiers*, in *Machine learning: ECML 2001*. 2001, Springer. p. 564-575.

[30] D.-Y. YEUNG and C. CHOW. "Parzen-window network intrusion detectors". *in Pattern Recognition, 2002. Proceedings. 16th International Conference on*. 2002. IEEE.