國立高雄科技大學資訊工程學系

基於網路封包偵測DDoS攻擊之研究

專題隊員：

|  |  |  |  |
| --- | --- | --- | --- |
| 姓名 | 學號 | E-mail | 負責項目說明 |
| 丁襄龍 | C109151163 | C109151163@  nkust.edu.tw | 收集資訊、文獻探討、理論設計、製作隨機森林模型、變數實驗 |
| 許哲晟 | C109151162 | C109151162@  nkust.edu.tw | 收集資訊、文獻研究、封包資訊研究、DDoS特徵值分析、製作訓練封包 |

中華民國 112年 6月 18日

1. 導論

1.1 研究動機

DDoS攻擊全名為Distributed Denial-of-Service Attack，又稱為分散式阻斷服務攻擊，是舊時DoS攻擊（Denial-of-Service Attack，阻斷服務攻擊）的擴大版。DDoS攻擊手法主要是製造大量網路流量泛洪特定目標，藉此以大量無效請求消耗目標的系統資源或網路頻寬，進而癱瘓其系統運作與服務。目前許多DDoS攻擊者利用「殭屍程式」植入多台電腦後門，組成「殭屍電腦」大軍之後，再發動大規模的DDoS泛洪流量。

近來網路攻擊頻繁發生，而DDoS更是常見的網攻手法。例如美國國會眾議院議長裴洛西（Nancy Pelosi）2022年訪台前夕，總統府及外交部等政府機關網站受到境外 DDoS 攻擊，其中總統府官網的流量高達平日的 200 倍，導致官網一度無法顯示[14]。另一個例子是俄羅斯最大的科技公司Yandex，也是俄羅斯第二大的搜尋引擎，於2021年9月遭受網路史上規模最大的、由全新殭屍網路Mēris發起的DDoS攻擊，其攻擊流量高達每秒2,180萬次。這波DDoS持續數周，橫掃紐西蘭、美國與俄羅斯各地。由於資安就是國安，我國政府特別把資安相關研究納入「六大核心戰略產業推動方案」之一[22]。有鑑於DDoS是近年來的資安重點，所以我們選擇DDoS作為我們專題研究的主題。

1.2 問題探討

我們研究的重點是做出一套偵測DDoS流量的系統，原因有以下三點:

1.我們認為防範 DDoS攻擊最重要的是即時且可靠的偵測出DDoS攻擊，並在裝置癱瘓前做出應對措施，讓損失降低。

2.目前許多企業使用雲端業者或ISP提供的流量清洗服務，但可能會因為清理中心與流量來源或伺服器相距遙遠，從而帶來流量延遲的問題。另外，把網路流量交由第三方清洗難免有資料外洩的風險。因此，最好的解決辦法即是在本地端偵測並做出應對措施。

3.根據文獻[23]軟體定義網路（SDN）遭受DoS\DDoS攻擊可能會被淹沒控制層、資料層，而控制層中的SDN控制器是整個網路架構的核心，主要負責管理資料層中的網路設備和應用層中的網路應用程序，所以如果是控制層被攻擊會導致整個網絡出現故障[24]；如果是資料層被 DoS/DDoS 攻擊可能會填滿 OpenFlow 交換器中的Flow Table。一旦交換器Flow Table已滿，交換器將無法存取從控制器接收的新的流程規則(flow rules)，且數據包也會被丟棄。

我們研究的DDoS偵測方法是偵測流進裝置的封包是否具有DDoS封包的特徵。舉例來說，SYN flood的封包時間上變化是否很小；是否收到的SYN 封包來自大量不同的source IP地址；是否收到的封包的source port為 80 (HTTP)；封包是否具有單一長度和窗口大小等。經由處理這些特徵值讓機器學習訓練出模型，以該模型為藍本來判斷封包流是否屬於DDoS攻擊封包，進而做出後續反應，如將封包引導到黑洞路由，並拒絕來源IP的再次發送封包給裝置。

另外，隨機森林演算法需要大量封包特徵值才能輸出較為準確的模組，因此在訓練過程中會耗費大量記憶體空間存取資料也會消耗系統效能；如果隨機森林中的決策樹個數很多時，訓練時所需要的時間要求會有所延長。因此在時效性要求很高的偵測任務時，需要加以改良。

1.3 問題重要性

網路已經成為我們日常生活中不可或缺的一部分，我們幾乎每天都會使用網路進行工作、學習、娛樂等活動。然而，隨著網路應用的不斷發展和普及，網絡攻擊也不斷增加，其中DDoS攻擊是最常見、最具破壞力的攻擊方式之一。

DDoS攻擊可以發送大量偽造的請求流量，使其超出正常處理能力範圍，從而導致網路堵塞、服務不可使用等問題，嚴重影響了用戶的網路體驗和伺服器的正常運行。而且，DDoS攻擊也可以掩蓋其他網路攻擊。例如數據竊取、惡意軟體傳播等，給網路安全和穩定帶來更大威脅。

因此，DDoS攻擊的檢測和防範顯得至關重要。只有及時發現並採取措施，才能有效遏制DDoS攻擊的發展，保障網路的安全和穩定。為了有效檢測和防範DDoS攻擊，我們除了需要提高網絡安全防範意識，加強網絡安全管理，配備專業的安全人員和設備，還需應用最新的技術和算法進行DDoS攻擊的檢測和防範。例如，可以使用機器學習算法、深度學習技術等，對網絡流量進行實時監控和分析，識別出異常流量和惡意請求，快速反應並採取相應的防禦措施。

綜上所述，DDoS攻擊對網路安全和穩定性構成了重大威脅，及時檢測和防範DDoS攻擊，是保障網路安全和穩定運行的重要措施。因此，我們需要不斷加強對DDoS攻擊的了解和研究，不斷創新和應用技術手段，以保障網路的安全和穩定。

1.4 文獻回饋

**多種機器學習比較**

在文獻[4]中使用3種機器學習方式Naive Bayes、KNN和隨機森林，來檢測和防止雲端服務器的DDoS攻擊，其中隨機森林效果最好，幾乎達到了99%的準確率，以及F1分數(一種判斷演算法精確度的指標)約為0.96。

文獻[6]使用隨機森林算法和Ryu框架做出「SDN DDoS攻擊早期檢測與緩解系統」，當受到DDoS攻擊時，經該團隊測試，檢測系統平均準確率為98.38%，平均檢測時間為36ms，平均緩解時間為1179ms，可以減少攻擊封包進入受害主機的平均數量，並且可以將SDN控制器上的平均CPU使用率減少44.9%。而該作者們使用的特徵值包含：

1.每個封包流的平均數據包(Average packets per flow)、

2.每個封包流的平均字節數(Average bytes per flow)、

3.不同通訊埠的增長量(Growth of different ports)、

4.每個區間的流表項數(The number of flow entries per interval)、

5.每個區間的來源IP數量(The number of source IPs per interval)，這些特徵值可供我們機器學習的參考。

Chen等人提出了一種使用隨機森林分類算法的方法[7]，主要關注OSI模型第4層協議的攻擊。通過從輸入數據流中來源IP、目標IP和目標通訊埠的組合獲得的不同熵信息，並用於訓練隨機森林分類器。將隨機森林分類器精度與隱馬爾可夫模型(HMM)和SVM分類器進行比較。結果表明，與HMM和SVM相比，隨機森林分類器具有最高的檢測率和最低的誤報率。作者們歸納出三種泛洪攻擊的特點：1.封包流的來源IP對於目標IP是多對一的關係、2.封包流的來源IP對於目標通訊Port是多對一的關係、3.封包流的目標Port和目標IP是多對一的關係，此特點可提供我們做參考。

在文獻[8]選取CIC-IDS-2017數據集中52萬個正常訪問的數據包作為實驗的正常流量。使用網絡安全工具hping3生成攻擊流量，並相應發出各種強度和類型的DDoS攻擊。對上述流量進行採集和提取特徵後生成數據集。共生成3652個樣本，其中正常流量樣本1433個，DDoS攻擊樣本2219個。訓練集和測試集按照7:3的比例劃分。使用支持向量機（SVM）和決策樹（DT）來比較和分析隨機森林算法的性能，隨機森林算法在測試集上的檢出率、準確率和誤報率分別為98.63%、98.99%和0.45%，整體性能優於決策樹（DT）和支持向量機（SVM）。

文獻[9]使用三種機器學習技術來檢測HTTP DDoS攻擊。包含KNN、Naïve Bayes 和隨機森林來檢測數據集CIDDS-001中的攻擊，Naïve Bayes的準確率結果計算為0.73，隨機森林分類器為0.999，KNN為0.992，而根據內文KNN和隨機森林分類器除了有高準確度也有著優異的性能表現。

根據以上文獻，小結為：近年來流行使用的機器學習演算法，包括 Naïve Bayes 、MLP、隨機森林、KNN、決策樹 (DT)、支持向量機 (SVM)，而根據特徵選擇、數據集選擇和所涉及的實現機制，仍然有利有弊，但從整體上來看使用隨機森林比其他算法能夠得到較高準確度。

**隨機森林與類神經網路比較**

根據文獻[10]的研究，目前大多數研究都在SDN中檢測高速率DDoS攻擊，而低速率DDoS攻擊由於難以檢測而具有很高的隱蔽性，需要單獨處理低速率DDoS攻擊，作者們使用一種基於遞歸神經網絡(RNN)的方法，該方法對攻擊進行了智能檢測，檢測準確率達到98.59%，有效阻擋了高速率DDoS攻擊。

根據文獻[11]J.Gojic和D.Radakovic提出了一種遞歸神經網路(RNN)的深度學習方法，用於檢測SDN網絡上的DDoS攻擊的模型。將RNN的模型與基礎設施層的Softmax回歸模型相結合，將網絡流量分類為惡意或正常，能成功地提高了所提出的5G移動網絡安全架構的安全性能，準確率高達99.83%。

文獻[12]主要目標是通過利用NCA-ANN來構建DDoS攻擊分類器模型，使用鄰里成分分析(NCA)的過程最小化了數據集的維度，再利用遞歸神經網路(RNN)的深度學習方法，經過10輪混淆矩陣，得出平均準確率為99.4％。

根據以上文獻，小結為：使用神經網路訓練出來的模型來偵測DDoS攻擊，準確率與隨機森林的相比無太大落差，但由於神經網路學習需要龐大的資料，導致訓練模型時需要大量的計算時間，所以「隨機森林演算法」有著時間成本低、迭代快的優勢相比之下是個較好的選擇。

1.5 比較分析

神經網路比隨機森林的準確度更勝一籌，為什麼不考慮使用神經網路？因為雖然神經網路，提供了很高的準確性，訓練模型時因需要大量的資料輔助，進而導致學習速度相當的慢，比其他類似的算法需要更多的時間來訓練，這就不符合我們的需求，當有大量DDoS攻擊時，我們所需要的是能給予及時偵測到是否為DDoS攻擊，加以防範我方流量爆炸，相比之下隨機森林能夠提供快速即時的反應，且準確度也差不多，是更適合拿來偵測DDoS攻擊，以下歸納出3個不選擇神經網路來當偵測DDoS攻擊的原因:

1.學習需要的資料量大

2.速度慢

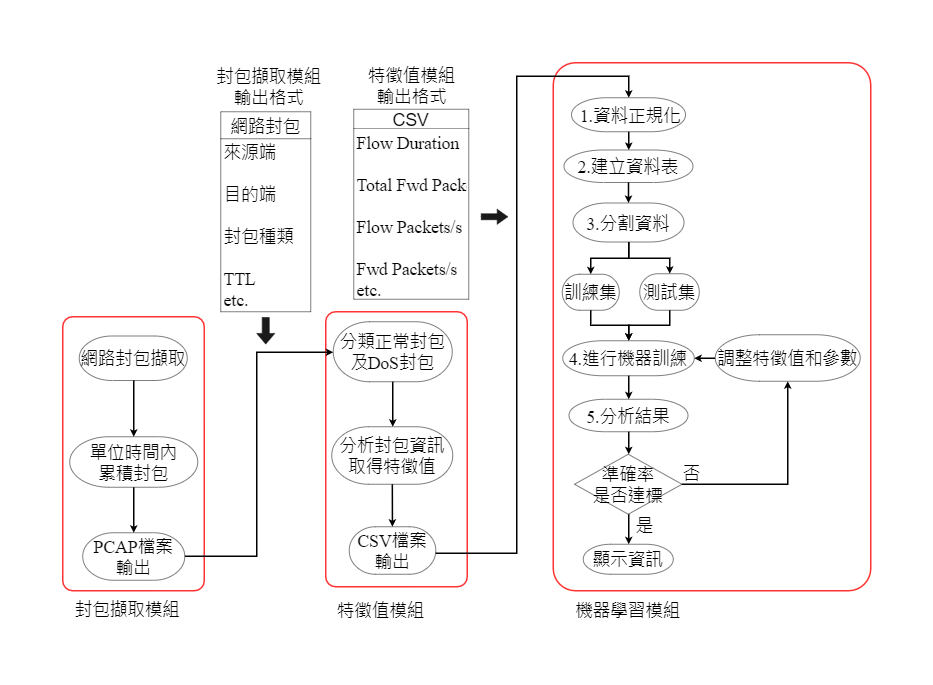
3.無法做到立即性

現有的研究仍然存在的共同問題：

如果面對剛出現且最新的DDoS攻擊技術，還是有可能會受到巨大的威脅和損失，因為新出現的DDoS攻擊的特徵值可能完全長得不一樣。

在現有的文獻調查，機器學習確實可以分辨出DDoS攻擊，可以有效緩解被DDoS攻擊的傷害程度，但之前被訓練好的模型在不久後，整體性能還能不能保持同樣的效果。近年，多種新型網路攻擊方式接續而來，能長時間有效阻擋DDoS攻擊是未知數，新型網路攻擊是否會造成巨大的傷害和損失。然而問題是大多的研究人員都使用舊有且僵化的數據集，因此與時俱進的取得新數據集非常重要，我們才可以檢測出新的DDoS攻擊。

1. 理論設計

2.1系統架構

圖一、封包擷取模組、特徵值模組及機器學習模組流程圖

1. 封包擷取模組:

封包擷取模組是捕捉網絡數據包的工具，能夠從網絡上抓取所有進出主機的數據包，包括通過網絡協議傳輸的文本、圖像、視頻和音頻等多媒體信息。

封包擷取模組中擷取器是Wireshark，能夠通過網絡介面監聽網絡流量，捕捉所有通過網絡介面的數據包，並輸出原始封包流的pcap檔。

1. 特徵值模組:

特徵值模組是用來對網路封包進行特徵提取的模組，使用各種算法來從封包數據中提取特徵，如統計特徵、流量特徵、頻譜特徵等等。特徵值模組需要考慮到提取特徵的效率和準確性，因為在高速網路中，封包的數量非常大，需要快速且準確地提取特徵才能應對實際應用場景。因此，特徵值模組的設計需要平衡提取特徵的準確性和效率。我們把從使用LOIC攻擊的來源IP所傳輸過來的封包歸類為DoS封包，並在該pcap檔中添加更多封包特徵值供機器學習模組做篩選。輸入為pcap檔，輸出為csv檔。

1. 機器學習模組:
2. 資料正規化，此步驟包含將缺失值或異常值移除或填補，缺失值指的是在資料集中某些欄位缺乏數值的情況，而異常值指的是在資料集中某些值明顯偏離正常分佈的情況。這些缺失值或異常值可能會影響資料分析和機器學習的準確性和可靠性，因此需要進行處理。再進行特徵轉換 (feature transformation) ，目的是將原始資料轉換為機器學習演算法可以使用的特徵表示，使得資料可以更好地被分析和處理。常見的特徵轉換方法包括特徵擷取、特徵選擇、特徵正規化、特徵標準化、特徵降維等。
3. 檢查完資料集中是否有缺失值、重複值、異常值等問題，接下來是特徵值選擇，目的是選擇最有用的特徵值建立模型。使用CICFlowMeter所得到的封包流之特徵值多達84種，我們將會根據文獻所提供的重要特徵值做篩選。
4. 分割資料（splitting data），是將樣本資料集切割成訓練集和測試集的過程，主要是為了評估和驗證機器學習模型的性能，以及防止模型過度擬合（overfitting）。可以按照一定的比例（例如7:3或8:2）將資料集隨機分成訓練集和測試集。通常，較大的資料集可以分配更多的樣本給訓練集，而較小的資料集可能需要分配更多的樣本給測試集
5. 進行隨機森林演算法，步驟如下：
   1. 隨機從訓練資料集中取樣，形成子資料集。這個過程稱為bootstrap取樣法（有放回地隨機選取樣本）。
   2. 從這個子資料集中選出K個特徵，進行樹的建立。K的選擇通常是根據經驗法則或者是通過交叉驗證的方式進行決定。
   3. 使用決策樹演算法建立一棵決策樹，該決策樹使用上述子資料集和選出的K個特徵進行訓練。
   4. 重複上述步驟，直到擁有足夠的決策樹。決策樹的數量可以通過交叉驗證的方式進行選擇。
   5. 對於每個新的樣本資料，使用每棵決策樹進行預測。多棵決策樹的預測結果可能不同，因此需要對預測結果進行聚合，例如投票，最終得到集成模型的預測結果。
6. 分析其模型，如果未符合準確度要求，就須調整特徵值和相關參數與權重，直到模型符合期望的準確度，將其輸出，關於分析的指標下單元會進行介紹。

2.2 DDoS特徵值與分析

關於DDoS特徵值，我們根據Reyhaneh Karimazad和Ahmad Faraahi在論文中[18]所提出的七個DDoS攻擊特徵辨識當發生DDoS攻擊時七項特徵會有明顯變化，並且擁有97%準確率，七個特徵代表意義如下所示：

|  |  |
| --- | --- |
| 1. Average Packet Size | 當受到攻擊時，封包平均大小將會在攻擊發生時急速上升。 |
| 2. Number of Packets | DDoS 攻擊會對目標主機發送大量的封包，因此，攻擊時的封包數量將會有明顯大於非攻擊時。 |
| 3. Time Interval Variance | 攻擊連線由程式統一發出，連線之間的間隔相當接近，因此攻擊間隔時間變異數在攻擊發生時將會趨近於 0。 |
| 4. Packet Size Variance | DDoS攻擊的惡意封包大小皆相同，一般情況下，即使是傳送相同檔案的封包大小也未必相同，因此當DDoS發生時，封包大小變異數將趨近於 0。 |
| 5. Number of Bytes | 在攻擊發生時，接受到的封包數和 Packet Size 將增加，因此該特徵會在攻擊發生時上升。 |
| 6. Packet Rate | 連線發送時間間隔短，這顯示著攻擊發生時 Packet Rate 將會相較於平常高。 |
| 7. Bits Rate | 每秒所接收到之 Bits 數，該特性非常高可以推斷出被DDoS攻擊。 |

表一、DDoS特特徵值(參考來源 [18])

本研究會統整表一[18]和文獻[6]中所提出的DDoS特徵值，判別那些特徵值對於我們機器學習有理想的成效，並給予那些特徵值相對的權重，目的是訓練出高準確度的模型。

根據先前文獻有關機器學習的比較，我們選擇使用隨機森林。而隨機森林是多棵決策樹結合而成，以「裝袋算法」為核心思想的模型，它使用重複抽樣的方法，將每一顆決策樹在不同的樣本基底或變數進行訓練，以降低單一模型的誤差。隨機森林具有準確性、簡單性、靈活性，可以用於分類和迴歸分析。另外，我們將通過使用「Gini算法」和「混淆矩陣」來驗證分類模型的準確性。Gini算法將多棵決策樹的結果統計分析，藉此隨機森林能夠獲得更高的精確度和更低的過擬合現象；混淆矩陣（Confusion Matrix）是一種用於評估分類器性能的工具，常用於二分類和多分類問題，有包含以下四種元素：

1. TP（True Positive）：

實際為True，預測為Positive。預測的結果正確(真陽性)

1. TN（True Negative）：

實際為True，預測為Negative。預測的結果正確(真陰性)

1. FP（False Positive）：

實際為False，預測為Positive。預測的結果錯誤(假陽性)

1. FN（False Negative）：

實際為False，預測為Negative。預測的結果錯誤(假陰性)



|  |  |  |  |
| --- | --- | --- | --- |
| 混淆矩陣 | | 實際結果(True Class) | |
| Positive | Negative |
| 模型預測  (Predicted Class) | Positive | **TP** | **FP** |
| Negative | **FN** | **TN** |

表二、混淆矩陣

可以根據表二得知該四種元素發生的情況。而這些元素可以拿來計算模型的準確率、精確率、召回率、F1指標，公式如下所示：

1. 準確率（Accuracy）=(TP+TN)/(TP+FP+FN+TN)，預測正確的樣本數除以總樣本數。但若類別不平衡，只以準確率來衡量模型表現會有偏差。
2. 精確率（Precision）= TP/(TP+FP)，在所有被模型預測為正的樣本中，真正為正的樣本數佔比。如精確率越高，表示模型的假陽性率（False Positive Rate）越低。
3. 召回率（Recall）= TP/(TP+FN)，即事實為真的樣本中有幾個是預測正確，有時候為了提高召回率，模型會過度標記陽性，從而增加了假陽性（False Positive）的數量，進而降低了模型的精度。
4. F1= 2 /[(1/Precision)+(1/Recall)]，即精確率與召回率的調和平均數，當精確率和召回率都很高時，F1 Score 也會較高。

2.3 機器學習演算法

四種機器學習：

**1.監督式學習 (Supervised Learning)**

提供大量的資料，這些資料當中，同時有資料和正解，供機器學習，要能夠訓練好機器，需要非常大量的資料，而且這些資料需要有明確的輸入與輸出的對應關係，這個方法需要我們先幫機器判斷出正確答案。

**2.半監督式學習 (Semi-supervised Learning)**

在半監督學習當中，我們使用少量的完整資料，以及其他不完整的資料來學習。可以減少對於完整資料的需求。

**3.非監督學習 (Unsupervised Learning)**

機器不需要完整的資料，機器自己透過演算法自己摸索出人類所期待的正確模型。

**4.強化學習 (Reinforcement Learning)**

在環境給予的獎懲刺激下，形成刺激的預期，來產生能夠獲得最大利益的習慣性行為，達到學習的效果，強調的是透過環境而行動，並會隨時根據輸入的資料逐步修正，類似人類的學習行為：透過與環境互動而獲得的回饋，來強化自己的學習。

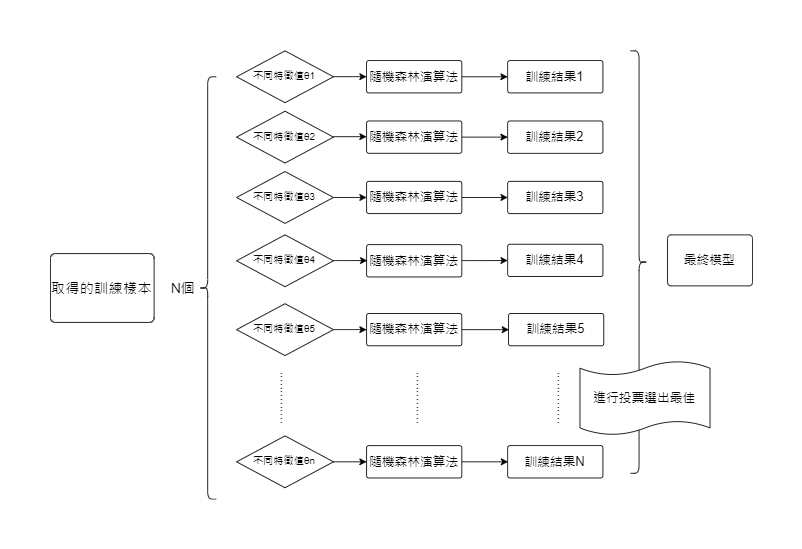
2.4 隨機森林介紹

由何天琴於 1995 年首次提出，她發明一個公式來使用隨機資料進行預測。之後在 2006 年，Leo Breiman 和 Adele Cutler 擴展了這個演算法，創造出我們今天所知的隨機森林。

隨機森林是一種機器學習演算法，隨機森林的中心思想是「裝袋法(Bagging)」，隨機抽取N個樣本，建立N棵決策樹。並進行重複抽樣的方法，讓每一顆樹都能在不同的樣本基底下獨立訓練，或者使用不同變數進行訓練，讓每一顆決策樹使用相異特徵值訓練，達到決策樹相互之間能夠截長補短之效果，最後綜合其預測結果，以有效降低以往單一模型可能造成的誤差，因此可以創造出一個具有廣泛多樣性，因為他的準確性、簡單性、靈活性，他可以用於分類和迴歸分析，加上其非線性特性，非常能夠適應各種資料和情況，受到許多資料科學家喜愛的模型。

2.5 基於隨機森林的DDoS演算法

當封包擷取模組累積到足夠數量時，便可以開始分析，先把資料先做基本的分類，像UTP、HTTP、TCP，不同攻擊方式有不同類型的特性，必須找到不同類型重要特徵值，並加以正規化並建成資料表，讓待會機器能夠看得懂我們傳輸的資料，之後分成資料集和訓練集進行訓練，訓練完成後再判斷資料集是否有受到攻擊，準確率是否良好，讓機器不停反覆地進行訓練，最終形成最好的模型，就可以用在實務上了。

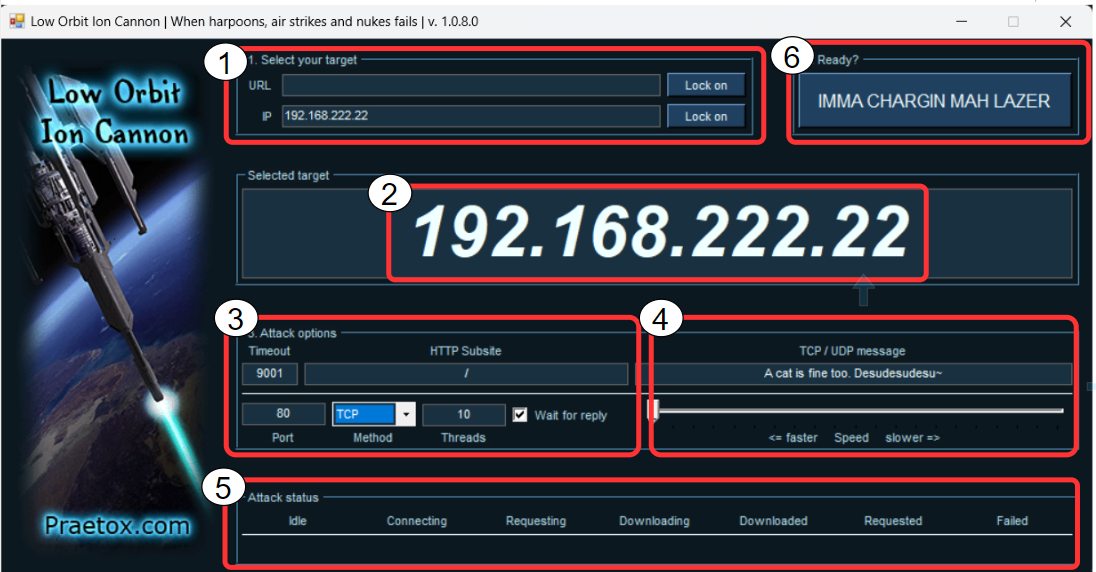
圖二、隨機森林演算法流程圖

隨機森林是多棵決策樹結合而成的，其中決策樹是使用 Gini 算法的決策樹，並選擇不同的訓練資料，生成的多棵決策樹，在訓練決策樹模型時，通常會以 Gini 指數為度量標準，選擇使 Gini 指數最小的劃分方式，模型具有較強的泛化能力，能準確地預測未知數據；反之，模型具有較弱的泛化能力，則可能會出現過擬合或欠擬合現象，無法準確地預測新數據。Gini 指數公式如下:

1. 製作方法
2. **流量產生**

使用隨機森林來偵測DDoS攻擊之前，需要先節錄大量網路封包當作訓練集，且在訓練集裡的封包不只是惡意封包，也需要有正常的流量封包。另外在訓練模型的過程中所使用的資料也需要有良好比例分配，這樣才能避免導致訓練出的隨機森林模型有過度擬合的狀況。若資料中數值太過發散可使用正規化技術，讓隨機森林訓練時能過更快的收斂。

(1)惡意封包:

我們使用LOIC來產生惡意封包，模擬有入侵者發送大量惡意流量封包至我方設備。LOIC是一種開源的網路攻擊工具，可以透過輸入目標設備的IP地址或網站的URL，來鎖定攻擊對象，並設置攻擊模式。共有三種攻擊模式(UTP、HTTP、TCP)可以選擇，之後可以設定攻擊時啟動多少個執行緒來控制發送封包的多寡、是否等待回復、傳送速率和數據包大小等參數，設定好之後即可點擊按鈕啟動攻擊，LOIC會生成大量流量並發送到目標網站，使其目標設備無法正常運作，進而讓合法用戶的訪問被丟包。圖三、LOIC操作介面

1. Select your target:輸入指定攻擊的目標設備的所在資訊，可以選擇網站的URL或IP地址來進行攻擊。
2. Select target:點選Lock on後，將會在此顯示攻擊的目標位址，讓攻擊者確定目標。因為使用LOIC攻擊是非法的，攻擊他人網站或系統可以導致廣泛的網路中斷，對受害者造成重大損失，會導致嚴重的刑事指控和民事訴訟。
3. Attack options:(1)Timeout可避免程式一直佔用系統資源而無法退出(2)HTTP Subsite 可以指定的是網站的子目錄或子頁面，(3)port可以指定攻擊的端口號，(4)Method可以設定攻擊模式，共有三種攻擊模式(UTP,HTTP,TCP) (5)Threads 指定使用的執行緒數。執行緒數越多，攻擊強度越大，但也會消耗更多的系統資源 (6)Wait for reply可以設定LOIC在向目標服務器發送網絡請求後，是否等待回復，可以避免服務器過載和崩潰的風險。
4. (1)TCP/UTP message 可指定發送到目標系統的消息。可以是空消息或自訂消息(2)可指定攻擊速度。速度越快，攻擊強度越大，但也會對目標系統造成更大的影響。
5. Attack status: LOIC的攻擊狀態，可以讓用戶了解到LOIC當前的攻擊類型和攻擊狀態，以便對攻擊進行控制和調整。同時，攻擊狀態也可以幫助用戶了解到LOIC是否在成功地攻擊目標服務器，以及攻擊是否需要進行調整或停止。
6. Ready attack:以上設定完之後，即可按下IMMA CHARGINMAH LAZER按鈕進行攻擊。

(2)正常封包:

正常流量封包是指符合標準網絡通信協議的網絡數據包。這些數據包由合法的來源地址和目的地址、端口號和協議類型組成，並遵循相應的協議規範。相比攻擊流量，正常流量的數據包通常具有更加穩定和可預測的特徵。我們的正常封包的來源為訪問合法網頁，如Google、Youtube等。

1. **特徵值萃取**

由上述步驟產生封包後，再使用CICFlowMeter把由Wireshark擷取的pcap檔轉錄成csv檔，不使用CICFlowMeter直接擷取的原因是CICFlowMeter擷取封包會因為攻擊封包流速度太快而出現丟包情形。但是CICFlowMeter提供了比Wireshark更豐富的特徵值可以做篩選，所以才要使用CICFlowMeter轉錄成csv檔案，而不是直接用Wireshark錄csv檔。CICFlowMeter是一種網絡流量監測和分析工具，可以用於識別網絡中的惡意流量，幫助網絡管理員保護網絡安全。CICFlowMeter可以對流量數據進行深度分析，該工具可以生成實時的流量統計信息，如流量大小、流量方向、流量持續時間。

現在，我們有著80多種特徵值的資料集，過多特徵值會使機器學習速度緩慢，因而先把一些不必要的特徵值給去除，如Flow ID、IP、時間戳記，再對資料進行整理，濾掉不完整封包，以防後續訓練隨機森林時造成異常。以下是經過文獻參考與多次訓練，對於DDoS攻擊有關聯，有以下7個特徵值:

* FlowDuration：指一個網絡流量的持續時間。
* TotalFwdPacket：指從源設備發往目的設備的封包總數。
* FwdPacketLengthMax：指所有從來源到目標封包中長度最長的一個。
* PacketLengthMean：指所有封包長度的平均值。
* FwdPackets/s：指在一秒內傳輸的從來源到目標封包數目。
* FlowBytes/s：指在一秒內傳輸的字節數。
* FwdSegmentSizeAvg：指所有從來源到目標區段尺寸的平均值。

1. **基於隨機森林的DDoS偵測**
2. 關鍵函式庫介紹

我們使用的程式有運用下列來製作機器學習:

* scikit-learn：scikit-learn是Python中最常用的機器學習庫之一，提供了包括分類、回歸、聚類、降維等在內的多種機器學習算法和相應的工具，如特徵選擇、交叉驗證、模型評估等。scikit-learn還提供了數據預處理和特徵提取等功能。
* numpy：numpy是Python中用於科學計算的基礎庫之一，提供了多維數組（ndarray）和矩陣操作等功能，支持向量化運算，可以大幅提高計算效率。numpy還提供了許多數學函數和隨機數生成器等功能。
* pandas：pandas是Python中用於數據分析和處理的資料庫之一，提供了高效的數據結構和數據操作函數，如數據讀取、過濾、合併、分組和重塑等功能。pandas主要支持的數據結構是Series（一維數組）和DataFrame（二維表格）。

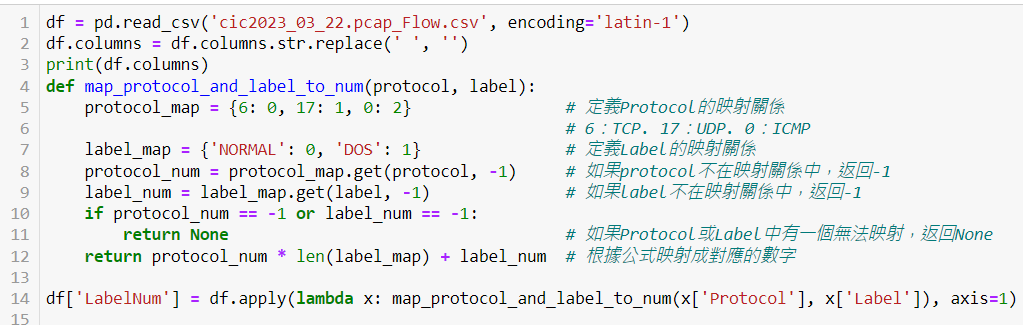
另外，我們使用下列函式庫來視覺化和分析模型結果:

* matplotlib.pyplot：matplotlib是Python中最常用的數據可視化函式庫之一，pyplot是matplotlib函式庫的一個模塊，提供了繪製各種圖形的函數。
* confusion\_matrix：為scikit-learn函式庫中的confusion\_matrix函數，用於計算分類模型的混淆矩陣。
* seaborn：seaborn是基於matplotlib的另一個數據可視化函式庫，提供了更多高級的數據可視化方式和風格設定。
* TSNE：為scikit-learn函式庫中的TSNE類別，用於高維數據的降維和可視化。而其中t-SNE（t-distributed stochastic neighbor embedding）是一種用於高維數據降維和可視化的非線性降維算法。相比於傳統的PCA和LDA等線性降維方法，t-SNE可以更好地保留原始數據的局部相似性和群集結構。t-SNE的基本思路是通過將高維數據映射到低維空間中，使得在原始空間中相似的數據點在映射後的低維空間中仍然相似，並且能夠更好地區分不同的群集。t-SNE通過一個相對較為複雜的公式來計算兩個數據點之間的相似度，並將相似度轉換為概率分布。然後，t-SNE在低維空間中通過最小化Kullback-Leibler（KL）相對熵的方法，使得映射後的低維數據點的概率分布和原始數據點的概率分布盡量接近，從而實現高維數據的降維和可視化。

1. **隨機森林步驟說明**

**第一步、檔案輸入**

首先載入CSV檔，再使用protocol和label交叉映射分辦出Normal\_\_TCP、DOS­\_TCP、Normal­\_UDP、DOS­\_UDP、Normal­\_ICMP封包，並將對應的數字存入新建的行標籤LabelNum，如圖四所示。



圖四、使用protocol和label交叉映射

**第二步:萃取特徵值和資料清洗**

 我們選擇特定的特徵值，這些特徵值在模型訓練和參考文獻中顯示對於模型的準確性很重要，並將它們存儲在一個名為packet\_chara\_value的DataFrame中。然後，我們使用fillna()方法將缺失值替換為0，以確保後續的數據分析和建模不受影響。接著，我們使用replace()方法將DataFrame中的正負無限大值替換為NaN，並最後使用dropna()方法刪除包含NaN值的行，以去除具有正負無限大值的DataFrame。

圖五、特徵值萃取及數據清洗

**第三步:分割資料**

把LabelNum分成y，其餘特徵值為x，將整個數據集分成訓練集和測試集，使訓練集占總數據集的70%，測試集占總數據集的30%，有x\_train、x\_test、y\_train、y\_test，以利於後面訓練隨機森林模型與測試模型。

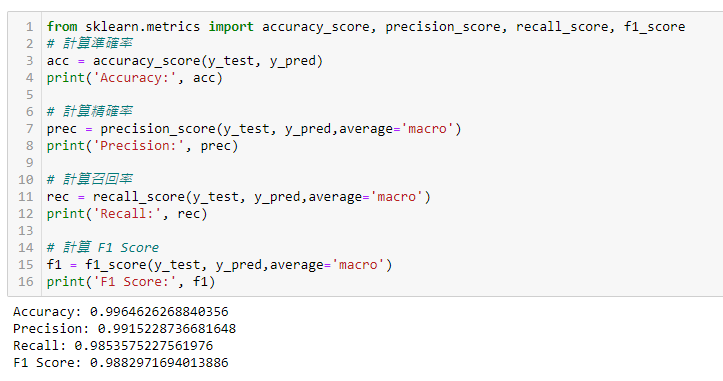
**第四步:隨機森林演算法及模型分數**

我們使用scikit-learn函式庫中的隨機森林分類器建立模型，參數如下：

* n\_estimators：隨機森林中決策樹的數量，這裡設置為100棵。
* max\_leaf\_nodes：每棵樹的最大葉子節點數，即是每棵樹的最大深度，這裡設置為32。
* oob\_score：是否計算袋外分數，這裡設置為True。
* random\_state：隨機種子，這裡設置為1。

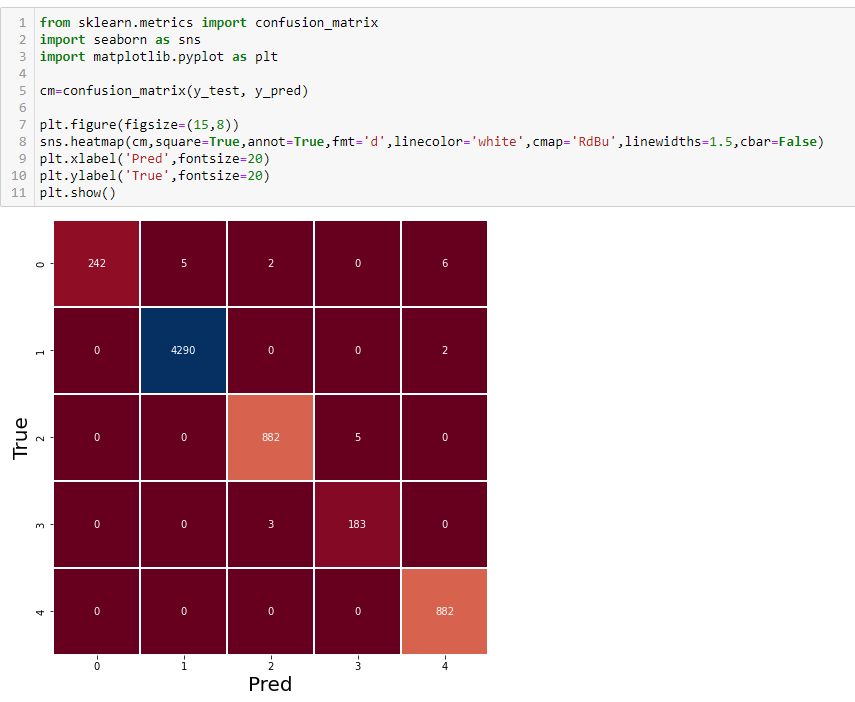
fit()方法的作用是使用訓練數據對模型進行訓練，從而得到一個 能夠對未知數據進行預測的模型。在fit()方法中，估計器將根據訓練數據調整其內部參數，使其能夠對未知數據進行正確的預測。在訓練完成後，估計器將存儲訓練好的模型，以便對新數據進行預測。最後，我們得出的訓練分數為99.6分、測試分數為99.5分、袋外分數為98.5分，可說是相當精確且沒有過度擬合的情形，其中袋外分數是樣本外準確度(out-of-bag score)指在模型訓練過程中使用袋外樣本來計算模型的準確度。因為有些樣本可能不會被用於訓練某些決策樹，這些樣本稱為「袋外樣本(out-of-bag samples)」。可以使用這些袋外樣本來估計模型在新數據上的準確度。

**第五步:** **模型預測結果與混淆矩陣分析**

使用scikit-learn套件中的metrics模組，如圖六，計算模型預測結果的準確率、精確率、召回率和F1分數，其中y\_test是測試集的實際標籤值，y\_pred是模型預測的標籤值，average 參數預設為binary，而目標標籤y\_test中包含多個類別，因此無法計算二元分類的精確率召回率和F1分數，所以需將average參數設定為macro即可對多個類別進行平均計算。

圖六、模型預測結果

我們使用confusion\_matrix()方法用於計算測試集之正解y\_test和預測之答案y\_pred兩個標籤向量之間的混淆矩陣。如圖七使用熱力圖所示，其中軸上刻度0代表Normal\_TCP、1代表DOS­\_TCP、2代表Normal­\_UDP、3代表DOS­\_UDP、4代表Normal­\_ICMP，y軸True代表正確解答，x軸Pred代表預測答案。而顏色代表其數量多寡，越紅代表數量少、反之數量越多。看的出來每行每列只有一格非深紅色，代表模型準度已相當精確，但在預測正解為Normal\_TCP有較高的預測錯誤率(5%)，有可能是所擷取之Normal\_TCP封包數量太少，我們將在下章節藉由調整參數及擷取更多Normal\_TCP封包盡可能提升Normal\_TCP的預測表現。



圖七、混淆矩陣熱力圖

1. 隨機森林演算法超參數實驗
2. **硬、軟體規格**
3. 硬體規格:

CPU規格:intel i5-6400

RAM:16G DDR4

硬碟:1TB SSD

1. 軟體規格

IDE:Jupyter Notebook v6.4.8

程式語言: Python 3.10.10

函式庫: Scikit-Learn v1.0.2、Numpy v1.21.5、Pandas v1.5.3、 matplotlib v3.5.1

輸入格式:.pcap、.CSV

1. **實驗目標與變數**

以下是使用隨機森林偵測DDoS的實驗變數及其定義：

實驗要測量的目標(應變數)：

* DDoS攻擊偵測的準確率，即隨機森林模型正確辨識DDoS攻擊封包的能力。

成本：

* 資源成本，訓練模型所需的時間。

操縱變數(自變數)：

* 樣本數量：訓練模型所用的樣本數量。
* 樹的數量：隨機森林中使用的決策樹數量。
* 樹的深度：決策樹的最大深度。
* 特徵數量：在每棵決策樹中使用的特徵數量。

控制變數(恆定不變)：

* 樣本集：使用相同的樣本集進行模型訓練和測試。
* 特徵集：使用相同的特徵集進行模型訓練和測試。
* 超參數：對於每一個超參數設置固定的值。
* 隨機種子：設置隨機種子來確保每次執行實驗得到相同的結果。

|  |  |  |
| --- | --- | --- |
| **變數** | **預設值** | **定義** |
| 目標（應變數） | 檢測準確度 | 評估隨機森林檢測DDoS攻擊的準確率 |
| 成本 | N/A | 時間成本 |
| 操縱變數（自變數） | 樣本量、樹的數量、樹的深度、特徵數量 | 分別改變隨機森林的樣本量、樹的數量、特徵數量等參數 |
| 控制變數（恆定不變） | 樣本集、特徵集、超參數、隨機種子 | 試驗中保持恆定不變的變數，包括訓練集和測試集的數據分佈、模型算法等 |

以下是隨機森林模型參數和檢測精度之間的關係：

|  |  |  |
| --- | --- | --- |
| **參數** | **定義** | **影響** |
| 樣本量 | 訓練集的樣本數 | 過少的樣本量可能導致模型的泛化能力不足，過多的樣本量可能導致訓練時間增加 |
| n\_estimators：  決策樹的數量 | 隨機森林中決策樹的數量 | 過少的樹的數量可能導致模型的不穩定性，過多的樹的數量可能導致訓練時間增加 |
| max\_depth：  決策樹的最大深度 | 隨機森林中決策樹的深度 | 會影響模型的複雜度和預測能力。如果樹的深度太深，模型可能會過度擬合訓練數據，導致在新數據上的表現較差 |
| max\_leaf\_nodes：  決策樹的最大葉節點數量 | 限制決策樹葉節點數量的參數 | 可以限制決策樹的複雜性，防止過度擬合。但較小的max\_leaf\_nodes 值可能會喪失一些複雜模式的捕捉能力。 |

在此實驗中，設定一個基準的隨機森林模型，選定一個參數為操作變數，依比例改變操縱變數，例如增加或減少樣本量、樹的數量或特徵數量，觀察這些參數對模型檢測精確度的影響。由下列步驟說明。

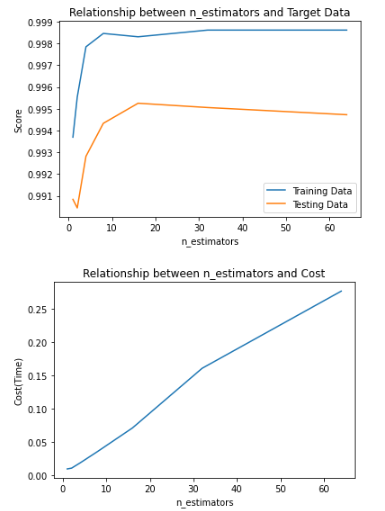
1. **操縱隨機森林某變數1之實驗，如圖八**

* 定義n\_estimators的範圍：將一系列的n\_estimators值定義為列表 n\_estimators = [1, 2, 4, 8, 16, 32, 64]。這些值將用作實驗中的n\_estimators參數。
* 初始化目標數據和成本列表：創建了三個空列表 target\_data\_train、target\_data\_test和cost\_data，分別用於存儲目標數據（準確率）和成本（訓練時間）。
* 遍歷n\_estimators範圍，進行實驗：使用for迴圈遍歷 n\_estimators列表中的每個值，對於每個n\_estimators值，進行以下操作：

1. 初始化隨機森林模型。
2. 紀錄開始時間。
3. 訓練隨機森林。
4. 紀錄結束時間。
5. 收集目標數據和成本。
6. **操縱隨機森林某變數2之實驗，**如圖九

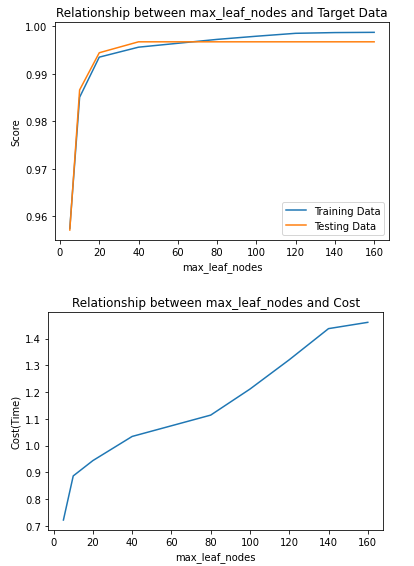
* 定義max\_leaf\_nodes的範圍：將一系列的max\_leaf\_nodes值定義為列表max\_leaf\_nodes = [5, 10, 20, 40, 80, 100, 120, 140, 160]。這些值將用作實驗中的max\_leaf\_nodes參數。
* 初始化目標數據和成本列表：創建了三個空列表target\_data\_train、target\_data\_test和cost\_data，分別用於存儲目標數據（準確率）和成本（訓練時間）。
* 遍歷max\_leaf\_nodes範圍，進行實驗：

1. 初始化隨機森林模型。
2. 紀錄開始時間。
3. 訓練隨機森林。
4. 紀錄結束時間。
5. 收集目標數據和成本。



圖八、決策樹的數量與模型分數、時間成本(秒)之間的關係圖

如圖八所示，其分數區間皆在0.99以上，隨然決策樹的數量在5棵之後模型分數就以微幅上升的趨勢，但其時間成本的增幅顯著。本測試以參數n\_estimators = 12當作最佳值。



圖九、最大葉節點數量與模型分數、時間成本(秒)之間的關係圖

如圖九，當max\_leaf\_nodes超過60時，隨然訓練集分數有些微上升，但測試集沒有上升跡象，所以決策樹葉節點數量超過60會開始發生過度擬合，因此我們選擇max\_leaf\_nodes = 60為最佳值

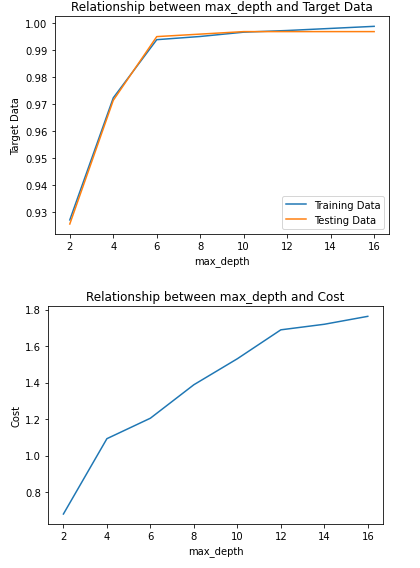
1. **操縱隨機森林某變數3之實驗，**如圖十

* 定義max\_depth的範圍：將一系列的max\_depth值定義為列表 max\_depth = [2, 4, 6, 8, 10, 12, 14, 16]。這些值將用作實驗中的max\_depth參數。
* 初始化目標數據和成本列表：創建了三個空列表 target\_data\_train、target\_data\_test和cost\_data，分別用於存儲目標數據（準確率）和成本（訓練時間）。
* 遍歷max\_depth範圍，進行實驗：使用for迴圈遍歷max\_depth列表中的每個值，對於每個max\_depth值，進行以下操作：

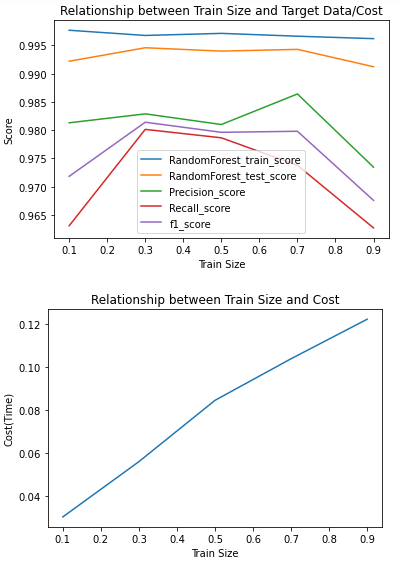
1. 初始化隨機森林模型。
2. 紀錄開始時間。
3. 訓練隨機森林。
4. 紀錄結束時間。
5. 收集目標數據和成本。
6. **操縱訓練資料量之實驗，如**圖十一

* 定義訓練集和目標變量：根據給定的特徵和標籤資料，將特徵資料存儲在x2中，將標籤資料存儲在y2中。
* 初始化目標數據和成本列表：創建了多個空列表rf\_score、rf\_score\_test、acc、prec、rec、f1和cost\_data，分別用於存儲目標數據（評估指標）和成本（訓練時間）。
* 初始化隨機森林模型：創建一個RandomForestClassifier隨機森林分類器，並設置n\_estimators、max\_leaf\_nodes、max\_depth、oob\_score和random\_state等參數。
* 遍歷train\_size範圍，進行實驗：使用for迴圈遍歷train\_size列表中的每個值，對於每個train\_size值，進行以下操作：

1. 分割訓練集和測試集
2. 紀錄開始時間
3. 訓練隨機森林
4. 紀錄結束時間
5. 預測測試集
6. 收集目標數據和成本

****圖十、決策樹的最大深度與模型分數、時間成本(秒)之間的關係圖

如圖十，當最大深度在12之後會開始有過度擬合跡象，而在深度為6時其上升幅度為最大，且時間成本相較低，因此我們選擇max\_depth=6為最佳值。

圖十一、訓練集資料量和各類分數、時間成本之間的關係圖

在此實驗中，經觀察在10%至30%測試分數和精確率（Precision）、召回率（Recall）、F1分數皆顯著上升；30%至70%訓練集分數沒太大起伏，而在70%時模型的預測正確性較高，但可能判斷錯一些真正的正例，因此召回率較低。當超過70%後其Recall（召回率）呈現下降趨勢，原因可能為:

1. 不平衡數據分布。
2. 過擬合現象

藉由F1分數把精確率和召回率平均，用來綜合評估模型的性能得出在訓練集佔30%時F1和訓練集分數有最高峰，因此在該比例有最好的模型分數。

**隨機森林中一棵決策樹結構**：

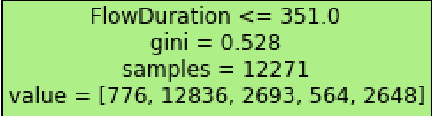
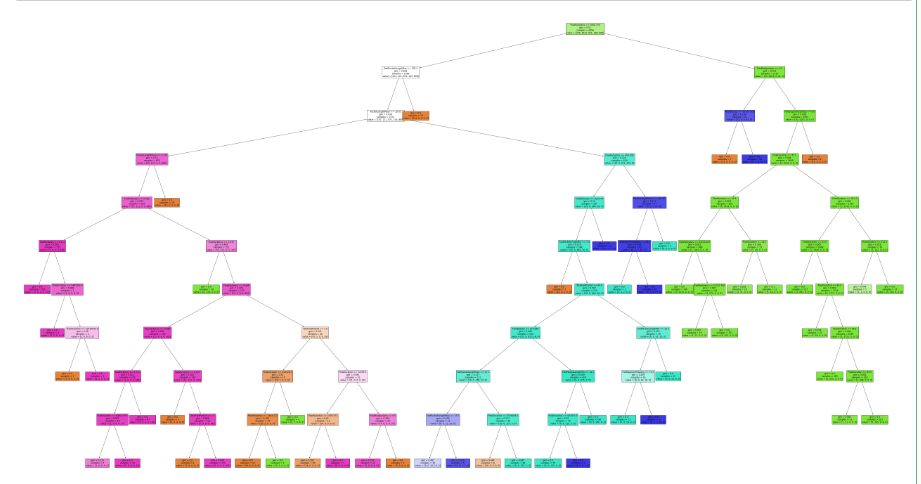
一棵決策樹的結構由節點（nodes）和邊（edges）組成。根節點（root）是整棵樹的起點，代表對整個資料集進行初始分割的條件。然後，從根節點開始，根據特徵的值進行分割，形成子節點（child nodes）。

每個節點包含以下信息：

1. 判斷條件：表示在該節點上進行的特徵判斷。
2. Gini指標：衡量節點純度的一個度量。
3. 樣本數量：表示在節點中包含的訓練樣本數量。
4. 類別值（value）：節點中每個類別的樣本數量。

決策樹的分支和結點連接形成一個樹狀結構，直到達到停止條件，例如達到最大深度、節點中的樣本數量小於某個閾值或無法再進行有效分割。在每個葉節點（leaf node）上，將給出最終的預測結果或類別分布。

隨機森林中的每棵決策樹都有其獨特的結構，通常比單獨的決策樹更深且更複雜。隨機森林通過組合多棵決策樹的預測，以增加模型的準確性。

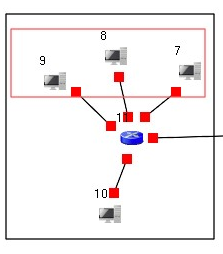


圖十二、隨機森林中一棵決策樹結構及根節點資訊

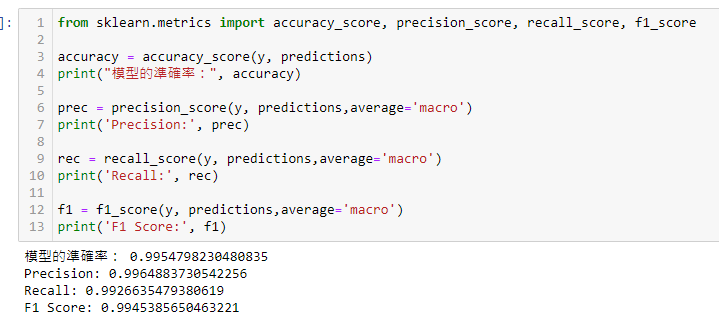
1. **成果**

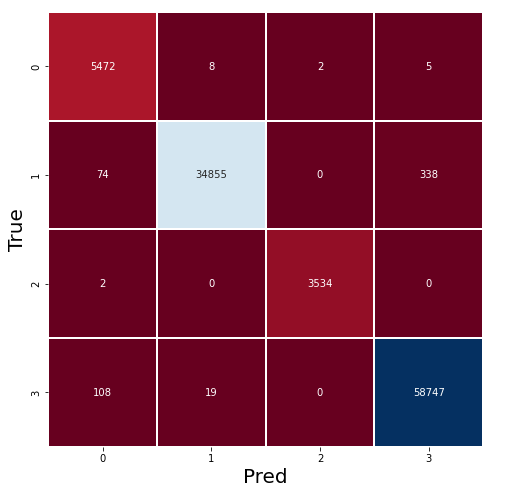
接下來，我們將使用Estinet分別在**傳統網路環境**和**SDN環境**擷取DDoS攻擊封包，包含Syn Flood、TCP Flood、UDP Flood，並進行使用前章節所歸納出的特徵值和超參數比例進行機器學習，製作出在各別環境的隨機森林演算法模型，此外，使用另外擷取封包流當作資料集來測試該隨機森林模型在實際應用上之準確性。

**[1]. 在傳統網路環境偵測Flood DDoS**

 使用Estinet模擬一個傳統的網路環境，架構如圖十三所示，主要針對TCP和UDP封包進行DDoS攻擊推測。攻擊者7、8、9號host通過11號路由器向10號接收端進行flood DDoS攻擊。我們在10號接收端使用TCPDUMP指令來收集封包再使用CICFlowMeter來把該Pcap檔解析成有更多特徵值的csv檔。接著，我們將使用預先訓練好的隨機森林模型來對CSV檔案中的封包進行預測，以區分它們是正常封包還是惡意封包。

圖十三、Estinet中的傳統的網路環境

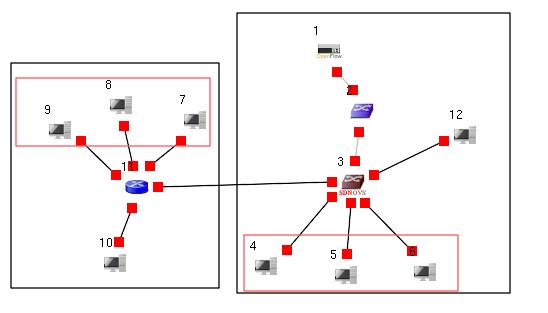
 我們使用和前章相同的超參數和特徵值在傳統虛擬網路環境中使用隨機森林演算法得出之模型訓練分數:99.5%、測試分數:99.4%。使用該模型來預測另一個在虛擬環境中擷取之資料集得到有99.5%的準確度，如圖十四所示，以此可以得出該模型沒有過度擬合，且可以實際運用到虛擬網路上。

圖十四、使用模型預測之準確度

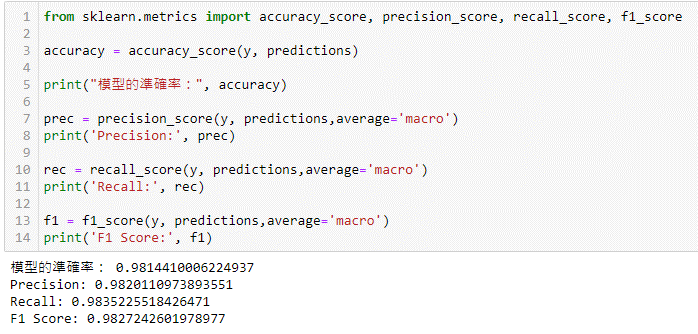
圖十五、測試傳統網路下的混淆矩陣熱力圖

圖十五為在Estinet中的傳統的網路環境下，使用隨機森林模型預測封包，所生成的混淆矩陣熱力圖，軸上刻度0為Normal­\_TCP、1為DOS\_TCP、2為Normal\_UDP、3為DOS­\_UDP，可以看出Normal­\_TCP預測正確率有比之前改善，但DOS\_TCP有部分被預測為DOS­\_UDP，有可能是因為DOS\_UDP類別的樣本數量比DOS\_TCP類別多，使模型傾向於預測該類別，或是因為資料集中可能存在標籤錯誤或雜訊資料。

**[2]. 在SDN虛擬環境偵測flood DDoS**

使用Estinet模擬一個傳統網路和**SDN**混和環境，架構如圖十六所示，主要針對TCP和UDP封包進行DDoS攻擊推測。攻擊者7、8、9號host通過11號路由器經由3號OpenFlowVSwitch向12號接收端進行flood DDoS攻擊；4、5、6號host通過3號OpenFlowVSwitch向12號接收端進行flood DDoS攻擊。我們在12號接收端使用TCPDUMP指令來收集封包再使用CICFlowMeter來把該Pcap檔解析成有更多特徵值的csv檔。接著，我們將使用預先訓練好的隨機森林模型來對CSV檔案中的封包進行預測，以區分它們是正常封包還是惡意封包。

圖十六、SDN和傳統網路混和架構圖

**** 我們使用和前章相同的超參數和特徵值在SDN和傳統網路混和環境中使用隨機森林演算法得出之模型訓練分數為98.1%、測試分數為98%。使用該模型來預測另一個在虛擬環境中擷取之資料集得到有98.1%的準確度，如圖十七所示，以此可以得出該模型沒有過度擬合，且可以實際運用到虛擬網路上。

圖十七、使用模型預測之準確度

1. **結論與未來工作**

**結論**

現今資安問題嚴重，不僅造成企業恐慌，也造成民眾困擾。本文的主要目的是在於如何快速且有效的偵測DDoS攻擊，對網路攻擊時的流量進行了研究，當遭到大量DDoS攻擊產生的網路封包與正常封包資訊的不同的特性，提出了使用隨機森林進行偵測DDoS攻擊，並把實驗結果輸出成直觀圖表。而在這過程中我們學到的知識如下

1. 了解DDoS攻擊的原理和常見攻擊方式，以及如何防範和應對。
2. 熟悉Wireshark和CICFlowMeter等網絡分析工具的使用，能夠對網絡流量進行分析和監測。
3. 學習特徵選擇和數據整理的方法，可以得到更加精確的特徵值集合，進一步提高DDoS攻擊的檢測效率和準確率。
4. 掌握機器學習算法的基本原理和應用，能夠運用機器學習技術對DDoS攻擊進行分類和檢測。

**未來工作**

1. 本文為偵測DDoS攻擊，但未做出後續處理，當偵測到大量DDoS攻擊可以透過其他工具(例如：SDN、Firewall)，使用SDN的控制器進行阻擋，當SDN控制器無法辨識的此封包，可以即時的對DDoS的傷害降到最小。
2. 當阻擋DDoS的功能完善後，可以開源供大眾使用。
3. DDoS偵測需要即時的處理，當DDoS封包過於龐大時，可能導致偵測延遲太高，單靠一個是不夠的，需要有效的分流，才不會錯失阻擋的時間，可能導致嚴重的後果。
4. 受限於電腦硬體，此實驗所使用之封包資料較少，未來可運用平行處理技術及更好設備使其所能容納更多樣也更龐大之數據集。
5. 增加更多新形態且更難以察覺的DDoS攻擊之判斷
6. 找出DDoS偵測系統的最佳配置，以在伺服器資源配置和流量允入控制之間實現最大淨利。透過優化DDoS偵測系統的伺服器資源配置，我們可以提高其效能，減少資源浪費，同時確保能夠應對潛在的DDoS攻擊。
7. 參考文獻
8. 劉煥彥 ,(2022年8月5日),yahoo新聞，台灣每月被境外網攻 3 千萬次！ DDoS 攻擊讓企業損失慘重，如何事先預防？, (https://tw.news.yahoo.com/news/台灣每月被境外網攻-3-千萬次-ddos-攻擊讓企業損失慘重-091733649.html)
9. Lubna Fayez Eliyan, Roberto Di Pietro,DoS and DDoS attacks in Software Defined Networks: A survey of existing solutions and research challenges,Future Generation Computer Systems,Volume 122,2021,Pages 149-171,ISSN 0167-739X,https://doi.org/10.1016/j.future.2021.03.011.(https://www.sciencedirect.com/science/article/pii/S0167739X21000911)
10. Ahmad, Ijaz & Namal, Suneth & Ylianttila, Mika & Gurtov, Andrei. (2015). Security in Software Defined Networks: A Survey. IEEE Communications Surveys & Tutorials. 17. 1-1. 10.1109/COMST.2015.2474118.
11. A. Mishra, B. B. Gupta, D. Peraković, F. J. G. Peñalvo and C. -H. Hsu, "Classification Based Machine Learning for Detection of DDoS attack in Cloud Computing," 2021 IEEE International Conference on Consumer Electronics (ICCE), 2021, pp. 1-4, doi: 10.1109/ICCE50685.2021.9427665.
12. Charles Griffiths. (2023,June 1).AAG It Services Ltd, "The Latest 2023 Cyber Crime Statistics (updated February 2023) , " (https://aag-it.com/the-latest-cyber-crime­-statistics/)
13. H. Nurwarsito and M. F. Nadhif, "DDoS Attack Early Detection and Mitigation System on SDN using Random Forest Algorithm and Ryu Framework," 2021 8th International Conference on Computer and Communication Engineering (ICCCE), 2021, pp. 178-183, doi: 10.1109/ICCCE50029.2021.9467167.
14. Y. Chen, J. Hou, Q. Li and H. Long, "DDoS Attack Detection Based on Random Forest," 2020 IEEE International Conference on Progress in Informatics and Computing (PIC), 2020, pp. 328-334, doi: 10.1109/PIC50277.2020.9350788
15. J. Tan, S. Jing, L. Guo and B. Xiao, "DDoS detection method based on Gini impurity and random forest in SDN environment," 2021 International Conference on Security, Pattern Analysis, and Cybernetics（SPAC), 2021, pp. 601-606, doi:
16. W. Al-gethami and A. Aljuhani, "Detection of HTTP Attacks using Machine Learning," 2022 2nd International Conference on Computing and Information Technology (ICCIT), 2022, pp. 344-348, doi: 10.1109/ICCIT52419.2022.9711638
17. M. W. Nadeem, H. G. Goh, Y. Aun and V. Ponnusamy, "A Recurrent Neural Network based Method for Low-Rate DDoS Attack Detection in SDN," 2022 3rd International Conference on Artificial Intelligence and Data Sciences (AiDAS), 2022, pp. 13-18, doi: 10.1109/AiDAS56890.2022.9918802.
18. J. Gojic and D. Radakovic, "Proposal of security architecture in 5G mobile network with DDoS attack detection," 2022 7th International Conference on Smart and Sustainable Technologies (SpliTech), 2022, pp. 1-5, doi: 10.23919/SpliTech55088.2022.9854338.
19. R. M. Abdullah, S. M. Abdullah and S. M. Abdullah, "Neighborhood Component Analysis and Artificial Neural Network for DDoS Attack Detection over IoT Networks," 2021 7th International Engineering Conference “Research & Innovation amid Global Pandemic" (IEC), 2021, pp. 1-6, doi: 10.1109/IEC52205.2021.9476123.
20. Karimazad, R., & Faraahi, A. (2011). An Anomaly-Based Method for DDoS Attacks Detection using RBF Neural Networks.