國立雲林科技大學

資訊管理系所

資料探勘專案作業報告

學生/學號：

廖帝登 D10823001

張　靖 M10823001

劉博茜 M10823015

盧廼安 M10823023

指導教授：許中川

中華民國 108 年 11 月

摘要

本研究針對汽車交易資料集和網路釣魚資料集，進行分類器的訓練和績效的預測，隨著大量資料的快速累積以及演算法與雲端運算的發展，大數據分析已經成為學術界與各行業關注與學習的焦點。從巨量資料中篩選有用的資訊，其運用的方法為資料探勘。本研究將使用 Python 撰寫 SVM 和 Adaboost 之程式進行模型訓練，並透過兩個分類器來比較，哪一個分類器在該資料集有比較好的績效。在 SVM 和 Adaboost 訓練之下，可以得知 SVM 在汽車交易資料集的分類準確度比網路釣魚資料集的績效來得較高。而 Adaboost 的分類準確度則在網路釣魚資料集的績效比汽車交易資料集的績效來得較高。從績效上的角度來看，可以知道 SVM 的分類在汽車交易資料集上，是比 Adaboost 來的要好；Adaboost 的分類在網路釣魚資料集的績效比 SVM 還要好。

關鍵字：資料探勘、Python、SVM、Adaboost、汽車交易、網路釣魚

目錄

第1章 緒論

1.1 研究背景與動機

本研究針對汽車交易資料集和網路釣魚資料集，進行分類器的訓練和績效的預測。汽車評估數據庫源自一個簡單的分層決策模型，該模型最初是為演示 DEX 而開發的，M. Bohanec, V。Rajkovic：決策專家系統。Sistemica 1（1），第 145-157 頁，1990。）。該模型根據以下概念結構評估汽車。然而網絡釣魚問題在“.COM”中被認為是至關重要的問題。行業，尤其是電子銀行和電子商務，涉及支付的在線交易數量。我們已經確定了與合法網站和網上誘騙網站相關的不同功能，並從不同來源收集了 1353 個不同網站。網絡釣魚網站是從 Phishtank 數據存檔（www.phishtank.com）收集的，該網站是一個免費的社區網站，用戶可以在其中提交，驗證，跟蹤並共享網絡釣魚數據。合法網站是使用 PHP 開發的網絡腳本從 Yahoo 和起點目錄中收集的。PHP 腳本已插入瀏覽器，我們從 1353 個網站中收集了 548 個合法網站。有 702 個網絡釣魚 URL 和 103 個可疑 URL。當一個網站被認為是可疑的時，這意味著該網站具有某些合法和虛假的功能。

1.2 研究目的

本研究主要以資料探勘為基礎，透過 Python 撰寫 SVM 和 Adaboost 之程式，以此進行模型訓練，這項研究針對汽車交易資料集和網路釣魚資料集，進行分類器的訓練和績效的預測。藉此了解兩者演算法在相同資料集的分類績效。

第2章 方法

2.1 程式架構- SVM

【SVM by car.data】

```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.svm import SVC
# 正規化
data_feature_name = ['buying','maint','doors','persons','lug_boot','safety','Class_Value']
#設定標籤
data =
pd.read_csv('D:\Python\DN_report2\car.data',header=None,names=data_feature_nam
e,sep = ',')
pattern_bying = {'vhigh':4,"high":3,"med":2,"low":1}
data['buying'] = [pattern_bying[x] if x in pattern_bying else x for x in data['buying']]
pattern_maint = {'vhigh':4,'high':3,'med':2,'low':1}
data['maint'] = [pattern_maint[x] if x in pattern_maint else x for x in data['maint']]
pattern_dorrs = {'2':2,'3':3,'4':4,'5more':5}
data['doors'] = [pattern_dorrs[x] if x in pattern_dorrs else x for x in data['doors']]
pattern_persons = {'2':2,'4':4,'more':6}
data['persons'] = [pattern_persons[x] if x in pattern_persons else x for x in
data['persons']]
pattern_lug_boot = {'small':1,'med':2,'big':3}
data['lug_boot'] = [pattern_lug_boot[x] if x in pattern_lug_boot else x for x in
data['lug_boot']]
pattern_safety = {'low':1,'med':2,'high':3}
data['safety'] = [pattern_safety[x] if x in pattern_safety else x for x in data['safety']]
pattern_class = {'unacc':1,'acc':2,'good':3,'vgood':4}
data['Class_Value'] = [pattern_class[x] if x in pattern_class else x for x in
data['Class_Value']]

# 切分訓練與測試資料
Y=data['Class_Value']
X = data.drop('Class_Value',axis=1)
X=X.to_numpy()
Y = Y.to_numpy()
```

```
X_train, X_test, y_train, y_test = train_test_split(X, Y, random_state=0,
train_size=0.8)
clf = SVC(kernel= 'linear',probability=True)
clf.fit(X_train,y_train)
clf.score(X_test,y_test)
print(clf.score(X_test,y_test))
```

【SVM by Fishing.data】

```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.svm import SVC
data_feature_name =
['SFH','popUpWidnow','SSLfinal_State','Request_URL','URL_of_Anchor','web_traffi
c','URL_Length','age_of_domain','having_IP_Address','Result']      #設定標籤
data =
pd.read_csv('D:\Python\DN_report2\PhishingData.data',header=None,names=data_fe
ature_name,sep = ',')
# 切分訓練與測試資料
Y = data['Result']
X = data.drop('Result',axis=1)
X=X.to_numpy()
Y = Y.to_numpy()
X_train, X_test, y_train, y_test = train_test_split(X, Y, random_state=0,
train_size=0.8)
# Adaboost
clf = SVC(kernel= 'linear',probability=True)
clf.fit(X_train,y_train)
clf.score(X_test,y_test)
print(clf.score(X_test,y_test))
```

2.2 程式架構-Adaboost

【Adaboost by car.data】

```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn import    ensemble, preprocessing, metrics
# 正規化
data_feature_name = ['buying','maint','doors','persons','lug_boot','safety','Class_Value']
#設定標籤
```

```
data =
pd.read_csv('D:\Python\DN_report2\car.data',header=None,names=data_feature_nam
e,sep = ',')
pattern_bying = {'vhigh':4,"high":3,"med":2,"low":1}
data['buying'] = [pattern_bying[x] if x in pattern_bying else x for x in data['buying']]
pattern_maint = {'vhigh':4,'high':3,'med':2,'low':1}
data['maint'] = [pattern_maint[x] if x in pattern_maint else x for x in data['maint']]
pattern_dorrs = {'2':2,'3':3,'4':4,'5more':5}
data['doors'] = [pattern_dorrs[x] if x in pattern_dorrs else x for x in data['doors']]
pattern_persons = {'2':2,'4':4,'more':6}
data['persons'] = [pattern_persons[x] if x in pattern_persons else x for x in
data['persons']]
pattern_lug_boot = {'small':1,'med':2,'big':3}
data['lug_boot'] = [pattern_lug_boot[x] if x in pattern_lug_boot else x for x in
data['lug_boot']]
pattern_safety = {'low':1,'med':2,'high':3}
data['safety'] = [pattern_safety[x] if x in pattern_safety else x for x in data['safety']]
pattern_class = {'unacc':1,'acc':2,'good':3,'vgood':4}
data['Class_Value'] = [pattern_class[x] if x in pattern_class else x for x in
data['Class_Value']]
# Adaboost
Y = data['Class_Value']
X = data.drop('Class_Value',axis=1)
X_train, X_test, y_train, y_test = train_test_split(X, Y, random_state=0,
train_size=0.8)
boost = ensemble.AdaBoostClassifier(n_estimators = 100,learning_rate=0.6)
boost_fit = boost.fit(X_train, y_train)
# 預測
test_y_predicted = boost.predict(X_test)
# 績效
accuracy = metrics.accuracy_score(y_test, test_y_predicted)
print(accuracy)
```

【Adaboost by Fishing.data】

```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn import    ensemble, preprocessing, metrics
from sklearn.svm import SVC
```

```
data_feature_name =
['SFH','popUpWidnow','SSLfinal_State','Request_URL','URL_of_Anchor','web_traffi
c','URL_Length','age_of_domain','having_IP_Address','Result']      #設定標籤
data =
pd.read_csv('D:\Python\DN_report2\PhishingData.data',header=None,names=data_fe
ature_name,sep = ',')
Y = data['Result']
X = data.drop('Result',axis=1)
X_train, X_test, y_train, y_test = train_test_split(X, Y, random_state=0,
train_size=0.8)
boost = ensemble.AdaBoostClassifier(n_estimators = 100,learning_rate=0.6)
boost_fit = boost.fit(X_train, y_train)
# 預測
test_y_predicted = boost.predict(X_test)
# 績效
accuracy = metrics.accuracy_score(y_test, test_y_predicted)
print(accuracy)
```

第3章 實驗

3.1 資料集

表格 1 汽車交易資料集

數據集特徵：	多變量	實例數：	1728	區域：	N/A
屬性特徵：	分類的	屬性數量：	6	貢獻日期	1997-06-01
相關任務：	分類	缺少價值？	No	網頁點擊數：	1074394

屬性名稱	屬性類別	類別種類	屬性說明
buying	Categorical	vhigh = 4 high = 3 med = 2 low = 1	買入價格
maint	Categorical	vhigh = 4 high = 3 med = 2 low = 1	維修保養價格

doors	Categorical	5more = 5	門數
persons	Categorical	more = 6	乘載人數
lug_boot	Categorical	small = 1 med = 2 big = 3	行李箱大小
safety	Categorical	low = 1 med = 2 high = 3	安全程度
class _values	Class Values	unacc = 1 acc = 2 good = 3 vgood = 4	接受程度

表格 1 網路釣魚資料集

數據集特徵：	多變量	實例數：	1353	區域：	電腦
屬性特徵：	整數	屬性數量：	10	貢獻日期	2016-11-02
相關任務：	分類	缺少價值？	N/A	網頁點擊數：	67614

服務器表單處理程序（SFH）：用戶提交信息後；該網頁會將信息傳輸到服務器，以便它可以對其進行處理。通常，信息是從加載網頁的同一域中處理的。網絡釣魚者採取措施使服務器表單處理程序為空，或者將信息轉移到與合法域不同的地方。

空白或為空 → 網絡釣魚

重定向到其他域 → 可疑

else → 合法的

使用彈出窗口：通常，經過身份驗證的網站不會要求用戶通過彈出窗口提交其憑據。

右鍵單擊禁用 → 網絡釣魚

顯示警報 → 可疑

else→合法的

偽造的 HTTPs 協議/ SSL final：每次傳輸敏感信息時，HTTPs 協議的存在反映出用戶確實與誠實的網站聯繫在一起。但是，網絡釣魚者可能使用偽造的 HTTPs 協議，從而可能欺騙用戶。因此，建議檢查 HTTPs 協議是否由受信任的發行者（例如 GeoTrust，GoDaddy，VeriSign 等）提供。

使用 https 和信任 issuer&年齡 >=2 年 → 合法的

不信任使用 https & issuer → 可疑

else → 網絡釣魚

請求網址：網頁通常由文字和一些對象（例如圖像和視頻）組成。通常，這些對像是從網頁的同一服務器加載到網頁中的。如果從 URL 地址欄中鍵入的域以外的域中加載對象，則該網頁可能具有可疑性。

請求網址 < 22% → 合法的

>= 22%和 < 61% → 可疑

else → 網絡釣魚

錨點的 URL ：與 URL 功能類似，但此處的網頁內鏈接可能指向與 URL 地址欄中鍵入的域不同的域。

URL 定位符% < 31% → 合法的

> =和<= 67% → 可疑

else → 網絡釣魚

網站訪問量：合法的網站通常會受到很高的訪問量，因為它們經常被訪問。由於網絡釣魚網站通常壽命較短。他們沒有網絡流量或排名較低。建議在 Alexa 數據庫中，合法網頁的排名小於或等於 150,000(Alexa，Web 信息公司，2011 年)。

Web 流量 < 150;000 →合法的

> 150；000！可疑

else → 網絡釣魚

長網址：網絡釣魚者隱藏網址的可疑部分，以重定向用戶提交的信息或將上載的頁面重定向到可疑域。從科學上講，沒有標準可靠的長度來區分網絡釣魚 URL 和合法 URL。(Mohammad et al。，2012）建議，當 URL 長度大於 54 個字符時，該 URL 可以被認為是虛假的。

URL 長度 < 54 → 合法的

>=54 和 <=75 → 可疑

else → 網絡釣魚

域名年齡：在線存在時間少於 1 年的網站可能被認為具有風險。

年齡 <= 6 個月 → 合法的

else → 網絡釣魚

IP 地址：在 URL 的域名中使用 IP 地址是表明有人試圖訪問個人信息的指示。此技巧涉及的鏈接可能以大多數公司不再使用的 IP 地址開頭。在前面進行的頻率分析中，20%的數據包含“IP”地址，並且所有這些都與犯罪網站相關聯。IP 地址類似於 http://91.121.10.211/~chems/webscr/verify 有時，該 IP 地址被轉換為十六進制，例如 http：//0x58.0xCC.0xCA.0x62。

URL 中存在 IP 地址 → 網絡釣魚

else → 合法的

類別：

合法 → 1

可疑 → 0

網絡釣魚 → -1

3.2 實驗設計

利用 python 進行實驗，透過原始數據(training data)進行 SVM、和 Adaboost 的訓練，藉此建立該數據的分類模型，透過測試數據(testing data)進行結果的預測分類，透過準確值的比較來了解，兩者分類器績效的好壞。

3.3 實驗結果

利用 python 進行 SVM、和 Adaboost 的程式撰寫，績效結果如下表。

	汽車交易資料集	網路釣魚資料集
SVM	0.82	0.78
Adaboost	0.78	0.80

第4章 結論

本研究的目的是利用 python 建構 SVM、和 Adaboost 在汽車交易資料集的分類模型；網路釣魚資料集的分類模型，在 SVM、和 Adaboost 訓練之下，可以得知 SVM 在汽車交易資料集的分類準確度比網路釣魚資料集的績效來得較高。而 Adaboost 的分類準確度則在網路釣魚資料集的績效比汽車交易資料集的績效來得較高。從績效上的角度來看，可以知道 SVM 的分類在汽車交易資料集上，是比 Adaboost 來的要好；Adaboost 的分類在網路釣魚資料集的績效比 SVM 還要好。

參考文獻

Abdelhamid, N., Ayesh, A., & Thabtah, F. (2014). Phishing detection based associative classification data mining. *Expert Systems with Applications*, *41*(13), 5948-5959.

Alexa, 2011. Alexa the Web Information Company (2011). *Alexa the Web Information Company.* <http://www.alexa.com/>.

Bohanec, M., & Rajkovič, V. (1990). DEX: An expert system shell for decision support. *Sistemica*, *1*(1), 145-157.

Mohammad, R. M., Thabtah, F., & McCluskey, L. (2012, December). An assessment of features related to phishing websites using an automated technique. In *2012 International Conference for Internet Technology and Secured Transactions* (pp. 492-497). IEEE.