Decision Tree for Intrusion Detection

Giacomo Ravara

11 Maggio 2018

1 Scopo del progetto

L'obiettivo del progetto é rappresentare, attraverso una Confusion Matrix, i risultati della classificazione e quindi del riconoscimento di fenomeni di Intrusion Detection da parte dellálgoritmo Decision Tree.

2 KDD Cup 1999 dataset

I database usati sono quelli della KDD Cup 1999. In particolare il train é stato fatto sul database 10% KDD dataset, che contiene circa 494,020 istanze. Ogni istanza é un vettore contenente i valori corrispondenti a 41 classi. Il testing é stato fatto sul database Corrected KDD, che include anche le corrette label identificative e che quindi ha permesso di realizzare la confusion matrix. É stato necessario effettuare un pre-processing sui dataset: sono stati convertiti i valori di alcune classi da testo o simbolo a interi.

3 Descrizione del lavoro

confusion_ matrix.py Questo file viene utilizzato nel progetto per implementare la confusion matrix. I vettori dati in ingresso alla confusion matrix vengono prima trasformati da una funzione che classifica i vari "attacchi" nelle 5 classi di interesse. Viene infine calcolato la percentuale di accuratezza.

plot.py Questo file implementa la funzione che viene utilizzata per realizzare la tabella della confusion matrix. Con l'opzione di avere i valori normalizzati.

decision_ tree.py In questo file viene caricato il dataset. Attraverso la libreria preprocessing di sklearn vengono modificati i valori di alcuni attributi. Viene poi effettuato l'apprendimento attraverso il classificatore di sklearn Decision Tree Classifier.

4 Conclusioni

Dall'analisi eseguita sul database Corrected KDDCup si ottengo i risultati che si attendevano. É possibile vedere che le connessioni Normal, DOS e Probing sono ben classificati. Non vale lo stesso per le connessioni R2L e U2R. Il risultato ottenuto é riscontrabile in Figura 1.

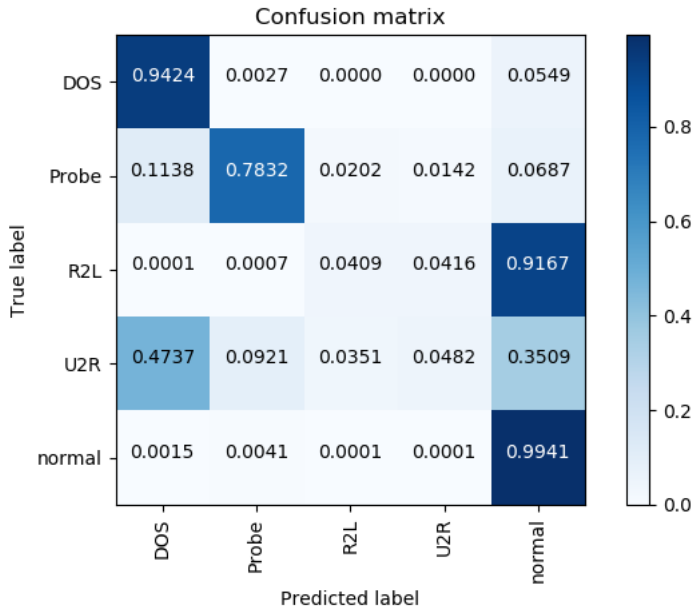

Figure 1: Confusion Matrix normalizzata

Nella seguente tabella é possibile vedere la percentuale di corretta classificazione (PCC), rispettivamente sul training dataset e poi sul test dataset.

	Training	**Testing**
PCC	99,99%	90,66%

5 Riproduzione Risultati

Per poter riprodurre i risultati presentati in questo progetto, sará necessario lanciare il file confusion_ matrix.py. Inoltre é necessario scaricare i file relativi al dataset KDD Cup 99. Per quanto riguarda la posssibilitá di poter scegliere la confusion matrix normalizzata, basta settare il rispettivo attributo in ingresso nella funzione del plot.