Feature Selection in the Corrected KDD -dataset

ZARGARI, Shahrzad <http://orcid.org/0000-0001-6511-7646>

Available from Sheffield Hallam University Research Archive (SHURA) at:
http://shura.shu.ac.uk/17048/

This document is the author deposited version. You are advised to consult the publisher's version if you wish to cite from it.

Published version


Copyright and re-use policy

See http://shura.shu.ac.uk/information.html
Feature Selection in the Corrected KDD-dataset

Shahrzad Zargari
Computing Department, Sheffield Hallam University
Contents

- Introduction
- Objective
- Methodology
- Experimental Work
- Conclusions
Introduction

Intrusion detection systems

Signature

Anomaly

Hybrid

Anomaly intrusion detection deals with detecting of unknown attacks in the network traffic, therefore, they are difficult to identify without human intervention. IT administrators struggle to keep up with Intrusion Detection System (IDS) alerts, and often manually examine system logs to discover potential attacks.

Final Goal

Automation of Intrusion detection by using data mining and statistical techniques
Objective

To propose a subset of features that can produce high intrusion detection rates while keeping the false positives at a minimum level. Therefore this will tackle the curse of dimensionality (e.g. reducing the computational complexity, time and power consumption).
The KDD CUP 1999\(^1\) is the first published dataset to be used in intrusion detection which has been used widely by researchers despite of the reported criticisms (McHugh, 2000) due to the lack of data.

The KDD-CUP 1999 dataset is a version of the dataset produced by the DARPA (1998) Intrusion Detection Evaluation Program which included nine weeks of raw TCP dump data for a local-area network (LAN) simulating a typical U.S. Air Force LAN. The LAN was operated as if it were a true Air Force environment, but peppered it with multiple attacks.

The full data: Kddcup.data.gz
A 10% subset: kddcup.data_10_percent.gz
The test data: kddcup.testdata.unlabeled.gz
Test data with corrected labels: correcte.gz

This study used the corrected test data for the data mining
The KDD-CUP 1999 Structure

- DOS: denial-of-service, e.g. syn flood
- Probing: surveillance and other probing, e.g., Port scanning
- R2L: Unauthorized access from a remote machine, e.g. guessing password
- U2R: Unauthorized access to local superuser (root) privileges, e.g., various “buffer overflow” attacks

The distribution of the attacks in the KDD-Cup 1999 dataset is different from the test KDD-Cup 1999 dataset.
The Features Proposal

Proposed Features

3) Service
5) Source bytes
6) Destination bytes
39) Dst host rerror rate
# The KDD-database Structure

## Features

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>G</th>
<th>H</th>
<th>I</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>duration</td>
<td>type</td>
<td>service</td>
<td>flag</td>
<td>src-byte</td>
<td>dst-byte</td>
<td>land</td>
<td>wrong-frg</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>tcp</td>
<td>http</td>
<td>S0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>tcp</td>
<td>http</td>
<td>S0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>G</th>
<th>H</th>
<th>I</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>duration</td>
<td>type</td>
<td>service</td>
<td>flag</td>
<td>src-byte</td>
<td>dst-byte</td>
<td>land</td>
<td>wrong-frg</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>tcp</td>
<td>http</td>
<td>S0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>tcp</td>
<td>http</td>
<td>S0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>G</th>
<th>H</th>
<th>I</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>duration</td>
<td>type</td>
<td>service</td>
<td>flag</td>
<td>src-byte</td>
<td>dst-byte</td>
<td>land</td>
<td>wrong-frg</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>tcp</td>
<td>http</td>
<td>S0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>tcp</td>
<td>http</td>
<td>S0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
The experimental Work

4 Samples from the Corrected KDD-CUP 1999 dataset

WEKA (V.3.7.4) Data mining software (Random Forest algorithm)

- Including all features
- Proposed features by this study (3,5,6, & 39)
- CfsSubsetEval + GreedyStepwise (2,3,5, & 6)
- InfoGainVal + Ranker (5,3,23, & 24)

Features suggested by this study (3,5,6, & 39) have higher intrusion detection rates with minimum false positives
Weka is a collection of machine learning algorithms for data mining tasks. Weka contains tools for data pre-processing, classification, regression, clustering, association rules, and visualization. It is also well-suited for developing new machine learning schemes.
A Typical Output of Weka

Classifier:
Choose RandomForest -I 10-K 0.51

Test options
- Cross-validation Folds 10
- Percentage split % 66

Classifier output:
- Correctly Classified Instances 90351, 94.3446 %
- Incorrectly Classified Instances 5416, 5.6554 %
- Kappa statistic 0.9234
- Mean absolute error 0.0034
- Root mean squared error 0.0417
- Relative absolute error 9.0131 %
- Root relative squared error 30.2045 %
- Coverage of cases (0.95 level) 99.6178 %
- Mean rel. region size (0.95 level) 3.021 %
- Total Number of Instances 95767

Detailed Accuracy By Class:

<table>
<thead>
<tr>
<th>Class</th>
<th>TP Rate</th>
<th>FP Rate</th>
<th>Precision</th>
<th>Recall</th>
<th>F-Measure</th>
<th>ROC Area</th>
</tr>
</thead>
<tbody>
<tr>
<td>apache2.</td>
<td>0.998</td>
<td>0</td>
<td>0.991</td>
<td>0.998</td>
<td>0.994</td>
<td>0.999</td>
</tr>
<tr>
<td>back.</td>
<td>0.999</td>
<td>0</td>
<td>0.999</td>
<td>0.999</td>
<td>0.999</td>
<td>0.999</td>
</tr>
<tr>
<td>buffer_overflow.</td>
<td>0.6</td>
<td>0</td>
<td>0.5</td>
<td>0.6</td>
<td>0.545</td>
<td>0.9</td>
</tr>
<tr>
<td>ftp_write.</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.5</td>
</tr>
<tr>
<td>guess_passwd.</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>imap.</td>
<td>0.086</td>
<td>0</td>
<td>0.692</td>
<td>0.086</td>
<td>0.153</td>
<td>0.978</td>
</tr>
<tr>
<td>http_tunnel.</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.5</td>
</tr>
<tr>
<td>ipsweep.</td>
<td>1</td>
<td>0</td>
<td>0.995</td>
<td>1</td>
<td>0.998</td>
<td>1</td>
</tr>
</tbody>
</table>
The experimental Work (2)

Including 10 features

4 Samples from the Corrected KDD-CUP 1999 dataset

WEKA (V.3.7.4) Data mining software (Random Forest algorithm)

Proposed features by this study (3,5,6,& 39)+ (4,14,16,27,28,& 37)

CfsSubsetEval + GreedyStepwise (2,3,5 & 6)+ (8,23,30,34,36,& 4)

InfoGainVal + Ranker (5,3,23 & 24)+ (33,35,2,36,34,& 6)

Feature set suggested by InfoGainVal+Ranker has higher intrusion detection rates however, comparing the results of applying only 4 features and 10 features, indicates that the detection rates improve slightly so it is a matter of trade off between increasing dimensionality or detection rate.
The result of data mining using different feature subsets

The results of data mining using three different subsets of features (4)

The results of data mining using three different subsets of features (10)
The same experimental work was carried out on NSL-KDD anomaly dataset. The results showed that the proposed features produce higher detection rates than the other two methods of data mining.

Conclusions

The statistical analysis of the Corrected KDD-CUP 1999 indicated that feature selection can reduce the high dimensions (curse of dimensionality) of the dataset and computational time while it does not have significant effect on intrusion detection rate.

The proposed subset of features (3,5,6,& 39) can be used in data mining tasks which performed better intrusion detections than the other subsets of features suggested by (CfsSubsetEval + GreedyStepwise) and (InfoGainVal + Ranker).

The subset of 10 features produced by InfoGainVal + Ranker algorithm performed better than the other subsets however, it is a matter of trade off (adding more dimensions) in order to improve the detection rate slightly.

The statistical analysis on NSL-KDD dataset confirmed the above results.

For future work, finding the optimum subset of features to be used in intrusion detection