Audit Data Reduction Using Neural Networks and Support Vector Machines

By

Srinivas Mukkamala, Andrew Sung

Presented At
The Digital Forensic Research Conference
DFRWS 2002 USA  Syracuse, NY (Aug 6th - 9th)

DFRWS is dedicated to the sharing of knowledge and ideas about digital forensics research. Ever since it organized the first open workshop devoted to digital forensics in 2001, DFRWS continues to bring academics and practitioners together in an informal environment. As a non-profit, volunteer organization, DFRWS sponsors technical working groups, annual conferences and challenges to help drive the direction of research and development.

http:/dfrws.org
Feature Ranking and Selection for Intrusion Detection using Support Vector Machines

Srinivas Mukkamala & Andrew H. Sung
Computer Science Department
New Mexico Tech
Intrusion Data

• Raw TCP/IP dump data collected from a network by simulating a typical U.S. Air Force LAN.
• For each TCP/IP connection, 41 various quantitative and qualitative features were extracted.
Attack Classes

Attacks fall into four main classes:

• Probing: surveillance and other probing.
• DOS: denial of service.
• U2R: unauthorized access to local super user (root) privileges.
• R2L: unauthorized access from a remote machine.
DARPA Data

**Attack Breakdown**

- smurf: 57.32215%
- neptune: 21.88491%
- normal: 19.85903%
- satan: 19.85903%
- ipsweep: 21.88491%
- portsweep: 0.25480%
- nmap: 0.25480%
- back: 0.25480%
- wasreclient: 0.25480%
- teardrop: 0.25480%
- pod: 0.25480%
- guess_passwd: 0.25480%
- land: 0.25480%
- loadmodule: 0.25480%
- ftp_write: 0.25480%
- multihop: 0.25480%
- phf: 0.25480%
- perl: 0.25480%
- spy: 0.25480%
- Other: 0.93391%
- satan: 0.32443%
- nmap: 0.04728%
- back: 0.04497%
- wasreclient: 0.04728%
- teardrop: 0.04728%
- pod: 0.04728%
- guess_passwd: 0.04728%
- land: 0.04728%
- loadmodule: 0.04728%
- ftp_write: 0.04728%
- multihop: 0.04728%
- phf: 0.04728%
- perl: 0.04728%
- spy: 0.04728%
DARPA Data

Attack Breakdown of 4898431 Attacks

Number of Instances
Support Vector Machines

• Learning systems that use a hypothesis space of linear functions in a high dimensional feature space.
• Trained with a learning algorithm from optimisation theory.
• Implements a hyperplane to perform a linear (2-class) separation.
Support Vector Classification

• Consider a 2 class problem

\[ F(x) = \begin{cases} -1: \text{class A} \\ +1: \text{class B} \end{cases} \]
The Feature Selection Problem

- Modeling an unknown function of a number of variables (features) based on data
- Relative significance of variables are unknown, they may be
  - Important variables
  - Secondary variables
  - Dependent variables
  - Useless variables
The Feature Selection Problem

• Which features are truly important?
• Difficult to decide due to:
  – Limited amount of data
  – Lack of algorithm
• Exhaustive analysis requires $2^n$ experiments (n = 41 in DARPA data).
• Need an empirical method.
Performance-Based Feature Ranking Method

• Delete one feature at a time.
• Use same training & testing sets (SVM & NN).
• If performance decreases, then feature is important.
• If performance increases, then feature is insignificant.
• If performance unchanges, then feature is secondary.
Performance-Based Feature Ranking: Procedure

• Compose the training and testing set; for each feature do the following
• Delete the feature from the training and the testing data;
• Use the resultant data set to train the classifier;
• Analyze the performance of the classifier using the test set, in terms of the selected performance criteria;
• Rank the importance of the feature according to the rules;
IDS Feature Ranking: Performance Factors

- Effectiveness.
- Training time.
- Testing time.
- False Positive Rate.
- False Negative Rate.
- Other relevant measures.
Feature Ranking: Sample Rules
Support Vector Machines

A (accuracy), LT (learning time), TT (testing time).

• If A \( \uparrow \) and LT \( \uparrow \) and TT \( \uparrow \), then feature is insignificant.
• If A \( \uparrow \) and LT \( \uparrow \) and TT \( \uparrow \), then feature is important.
• If A \( \uparrow \) and LT \( \uparrow \) and TT \( \uparrow \), then feature is important.
• Otherwise, feature is secondary.
Feature Ranking: Sample Rules
Neural Networks

A (accuracy), FP (false positive rate), FN (false negative rate).

• If $A \uparrow$ and $FP \uparrow$ and $FN \uparrow$, then feature is insignificant.
• If $A \uparrow$ and $FP \uparrow$ and $FN \uparrow$, then feature is important.
• If $A \uparrow$ and $FP \uparrow$ and $FN \uparrow$, then feature is important.
  
• Otherwise, feature is secondary.
Rule Set

1. If accuracy decreases and training time increases and testing time decreases, then the feature is important.

2. If accuracy decreases and training time increases and testing time increases, then the feature is important.

3. If accuracy decreases and training time decreases and testing time increases, then the feature is important.

4. If accuracy unchanged and training time increases and testing time increases, then the feature is important.

5. If accuracy unchanged and training time decreases and testing time increases, then the feature is secondary.
Rule Set

6. If accuracy unchanges and training time increases and testing time decreases, then the feature is secondary

7. If accuracy unchanges and training time decreases and testing time decreases, then the feature is unimportant

8. If accuracy increases and training time increases and testing time decreases, then the feature is secondary

9. If accuracy increases and training time decreases and testing time increases, then the feature is secondary

10. If accuracy increases and training time decreases and testing time decreases, then the feature is unimportant
Performance-Based Feature Ranking

Advantages

• General applicability (ANNs, SVMs, etc.)
• Linear complexity (requiring only $O(n)$ experiments).
• Tuning of rules to improve results.
• Multi-level ranking is possible.
Performance-Based Feature Ranking Results

<table>
<thead>
<tr>
<th></th>
<th>Important</th>
<th>Secondary</th>
<th>Unimportant</th>
</tr>
</thead>
<tbody>
<tr>
<td>Normal</td>
<td>1,3,5,6,8-10,14,15,17,20-23,25-29,33,35,36,38,39,41,2,4,7,11,12,16,18,19,24,30,31,34,37,40,13,32</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Probe</td>
<td>3,5,6,23,24,32,33,1,4,7-9,12-19,21,22,25-28,34-41,2,10,11,20,29,30,31,36,37</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DOS</td>
<td>1,3,5,6,8,19,23-28,32,33,35,36,38-41,2,7,9-11,14,17,20,22,29,30,34,37,4,12,13,15,16,18,19,21,31</td>
<td></td>
<td></td>
</tr>
<tr>
<td>U2R</td>
<td>5,6,15,16,18,25,32,33,7,8,11,13,17,19-24,26,30,36-39,9,10,12,14,27,29,31,34,35,40,41</td>
<td></td>
<td></td>
</tr>
<tr>
<td>R2L</td>
<td>3,5,6,24,32,33,2,4,7-23,26-31,34-41,1,20,25,38</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
# SVM: Using All 41 Features

<table>
<thead>
<tr>
<th>Class</th>
<th>Training time (sec)</th>
<th>Testing time (sec)</th>
<th>Accuracy</th>
<th>Class size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Normal</td>
<td>7.66</td>
<td>1.26</td>
<td>99.55%</td>
<td>1000:1400</td>
</tr>
<tr>
<td>Probe</td>
<td>49.13</td>
<td>2.10</td>
<td>99.70%</td>
<td>500:700</td>
</tr>
<tr>
<td>DOS</td>
<td>22.87</td>
<td>1.92</td>
<td>99.25%</td>
<td>3002:4207</td>
</tr>
<tr>
<td>U2R</td>
<td>3.38</td>
<td>1.05</td>
<td>99.87%</td>
<td>27:20</td>
</tr>
<tr>
<td>R2L</td>
<td>11.54</td>
<td>1.02</td>
<td>99.78%</td>
<td>563:563</td>
</tr>
</tbody>
</table>
## SVM: Using Important Features

<table>
<thead>
<tr>
<th>Class</th>
<th>No of Features</th>
<th>Training time (sec)</th>
<th>Testing time (sec)</th>
<th>Accuracy</th>
<th>Class size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Normal</td>
<td>25</td>
<td>9.36</td>
<td>1.07</td>
<td><strong>99.59%</strong></td>
<td>5092:6890</td>
</tr>
<tr>
<td>Probe</td>
<td>7</td>
<td>37.71</td>
<td>1.87</td>
<td><strong>99.38%</strong></td>
<td>1000:1400</td>
</tr>
<tr>
<td>DOS</td>
<td>19</td>
<td>22.79</td>
<td>1.84</td>
<td><strong>99.22%</strong></td>
<td>3002:4207</td>
</tr>
<tr>
<td>U2R</td>
<td>8</td>
<td>2.56</td>
<td>0.85</td>
<td><strong>99.87%</strong></td>
<td>27:20</td>
</tr>
<tr>
<td>R2L</td>
<td>6</td>
<td>8.76</td>
<td>0.73</td>
<td><strong>99.78%</strong></td>
<td>563:563</td>
</tr>
</tbody>
</table>
### SVM: Using Union of Important Features of All Classes, 30 Total

<table>
<thead>
<tr>
<th>Class</th>
<th>Training time</th>
<th>Testing time</th>
<th>Accuracy</th>
<th>Class size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Normal</td>
<td>7.67</td>
<td>1.02</td>
<td>99.51%</td>
<td>1000:1400</td>
</tr>
<tr>
<td>Probe</td>
<td>44.38</td>
<td>2.07</td>
<td>99.67%</td>
<td>500:700</td>
</tr>
<tr>
<td>DOS</td>
<td>18.64</td>
<td>1.41</td>
<td>99.22%</td>
<td>3002:4207</td>
</tr>
<tr>
<td>U2R</td>
<td>3.23</td>
<td>0.98</td>
<td>99.87%</td>
<td>27:20</td>
</tr>
<tr>
<td>R2L</td>
<td>9.81</td>
<td>1.01</td>
<td>99.78%</td>
<td>563:563</td>
</tr>
</tbody>
</table>
SVM: Using Important Features + Secondary Features

<table>
<thead>
<tr>
<th>Class</th>
<th>No of Features</th>
<th>Training time (sec)</th>
<th>Testing time (sec)</th>
<th>Accuracy</th>
<th>Class size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Normal</td>
<td>39</td>
<td>8.15</td>
<td>1.22</td>
<td>99.59%</td>
<td>1000:1400</td>
</tr>
<tr>
<td>Probe</td>
<td>32</td>
<td>47.56</td>
<td>2.09</td>
<td>99.65%</td>
<td>500:700</td>
</tr>
<tr>
<td>DOS</td>
<td>32</td>
<td>19.72</td>
<td>2.11</td>
<td>99.25%</td>
<td>3002:4207</td>
</tr>
<tr>
<td>U2R</td>
<td>25</td>
<td>2.72</td>
<td>0.92</td>
<td>99.87%</td>
<td>27:20</td>
</tr>
<tr>
<td>R2L</td>
<td>37</td>
<td>8.25</td>
<td>1.25</td>
<td>99.80%</td>
<td>563:563</td>
</tr>
</tbody>
</table>
Performance Statistics
(using performance-based ranking)

- All features
- Important features + Secondary features
- Important features
- Union of important features
## Performance Statistics
(Using performance-based ranking)

<table>
<thead>
<tr>
<th></th>
<th>Normal</th>
<th>Probe</th>
<th>DOS</th>
<th>U2R</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>%</strong></td>
<td>99.59%</td>
<td>99.59</td>
<td>99.55</td>
<td>99.51</td>
</tr>
<tr>
<td><strong>%</strong></td>
<td>99.70</td>
<td>99.67</td>
<td>99.65</td>
<td>99.38</td>
</tr>
<tr>
<td><strong>%</strong></td>
<td>99.25</td>
<td>99.25</td>
<td>99.22</td>
<td>99.22</td>
</tr>
<tr>
<td><strong>%</strong></td>
<td>99.87</td>
<td>99.87</td>
<td>99.87</td>
<td>99.87</td>
</tr>
<tr>
<td><strong>%</strong></td>
<td>99.80</td>
<td>99.78</td>
<td>99.78</td>
<td>99.78</td>
</tr>
</tbody>
</table>
Feature Ranking using Support Vector Decision Function

$$F(X) = \left\lvert W_i X_i + b \right\rvert$$

- $F(X)$ depends on the contribution of $W_i X_i$
- Absolute value of $W_i$ measures the strength of classification of classification
Feature Ranking using Support Vector Decision Function (SVDF)

• *if* $W_i$ is a large positive value *then* the $i^{th}$ feature is a key factor for the positive class

• *if* $W_i$ is a large negative value *then* the $i^{th}$ feature is a key factor for the negative class

• *if* $W_i$ is a value close to zero on either the positive or negative side *then* the $i^{th}$ feature does not contribute significantly to the classification
SVM Based Feature Ranking Method

- Calculate the weights from the support vector decision function.
- Rank the importance of the features by the absolute values of the weights.
- Delete the insignificant features from the training and the testing data.
- Use the resultant data set to train the classifier.
- Analyze the performance of the classifier using the test set, in terms of the selected performance criteria (threshold values of the weights for ranking the features).
SVM Based Feature Ranking: Advantages

- Uses SVMs decision function.
- Linear complexity (requiring only $O(n)$ experiments).
- Tuning of the ranking process by adjusting the threshold values.
- Multi-level ranking is possible.
## SVM-Based Feature Ranking Results

### Important

<table>
<thead>
<tr>
<th>Normal</th>
<th>2,3,4,6,10,12,23,29,32,33,34,36, 1,5,7-9,11,13-22, 24-28,30,31,35,37-41</th>
</tr>
</thead>
<tbody>
<tr>
<td>Probe</td>
<td>2,4,5,23,24,33, 1,3,6-22,25-32,34-41</td>
</tr>
<tr>
<td>DOS</td>
<td>23,24,25,26,36,38,39, 1-22,27-35,40,41</td>
</tr>
<tr>
<td>U2R</td>
<td>1,2,4,5,12,29,34, 3,6-11,13-28,30-33,35-41</td>
</tr>
<tr>
<td>R2L</td>
<td>1,3,32, 2,4-31,33-41</td>
</tr>
</tbody>
</table>
### SVM: Using Important Features as ranked by SVDF

<table>
<thead>
<tr>
<th>Class</th>
<th>No of Features</th>
<th>Training time (sec)</th>
<th>Testing time (sec)</th>
<th>Accuracy</th>
<th>Class size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Normal</td>
<td>15</td>
<td>3.73</td>
<td>.98</td>
<td>99.56%</td>
<td>1000:1400</td>
</tr>
<tr>
<td>Probe</td>
<td>12</td>
<td>41.44</td>
<td>1.63</td>
<td>99.35%</td>
<td>500:700</td>
</tr>
<tr>
<td>DOS</td>
<td>16</td>
<td>20.43</td>
<td>1.62</td>
<td>99.14%</td>
<td>3002:4207</td>
</tr>
<tr>
<td>U2Su</td>
<td>13</td>
<td>1.82</td>
<td>0.97</td>
<td>99.87%</td>
<td>27:20</td>
</tr>
<tr>
<td>R2L</td>
<td>6</td>
<td>3.24</td>
<td>.98</td>
<td>99.72%</td>
<td>563:563</td>
</tr>
</tbody>
</table>
### SVM: Union of Important Features of All Classes: 19 Total

Training : testing  5092 : 6890

<table>
<thead>
<tr>
<th>Class</th>
<th>Training time</th>
<th>Testing time</th>
<th>Accuracy</th>
<th>Class size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Normal</td>
<td>4.35</td>
<td>1.03</td>
<td>99.55%</td>
<td>1000:1400</td>
</tr>
<tr>
<td>Probe</td>
<td>26.52</td>
<td>1.73</td>
<td>99.42%</td>
<td>500:700</td>
</tr>
<tr>
<td>DOS</td>
<td>8.64</td>
<td>1.61</td>
<td>99.19%</td>
<td>3002:4207</td>
</tr>
<tr>
<td>U2R</td>
<td>2.04</td>
<td>0.18</td>
<td>99.85%</td>
<td>27:20</td>
</tr>
<tr>
<td>R2L</td>
<td>5.67</td>
<td>1.12</td>
<td>99.78%</td>
<td>563:563</td>
</tr>
</tbody>
</table>
Performance Statistics
(using SVM-based ranking)

- All features
- Important features
- Union of important features
<table>
<thead>
<tr>
<th></th>
<th>Normal</th>
<th>Probe</th>
<th>DOS</th>
<th>U2R</th>
<th>R2L</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>%</strong></td>
<td>99.56%</td>
<td>99.55</td>
<td>99.55</td>
<td>99.87</td>
<td>99.78</td>
</tr>
<tr>
<td><strong>%</strong></td>
<td>99.87</td>
<td>99.87</td>
<td>99.85</td>
<td>99.78</td>
<td>99.78</td>
</tr>
</tbody>
</table>
IDS Feature Ranking: Performance Factors

- Effectiveness.
- Training time.
- Testing time.
- False Positive Rate.
- False Negative Rate.
- Other relevant measures.
Two Feature Ranking Methods: Performance Summary

- Important features selected by two methods heavily overlap.
- Different levels of SVM IDS performance are achieved by
  - using all features
  - using important features
  - using union of important features
- However, the performance difference is small
A New IDS Architecture Using SVMs

IDS

SVM 1 (Normal)
SVM 2 (Probe)
SVM 3 (DOS)
SVM 4 (U2Su)
SVM 5 (R2L)

Flag ?

System Administrator

Servers

Internet

Preprocessor

Firewall

Servers
Conclusions

• IDS based on SVMs.
• SVMs generally outperform NNs (cf. reference 2)
• Two methods for feature ranking of 41 inputs, for each of the 5 classes.
• Using important features give comparable performance.
• New IDS comprising 5 SVMs delivers high accuracy and faster (than NN) running time.
References

