An Autonomous Labeling approach to Support Vector Machines Algorithms for Network Traffic Anomaly Detection

Carlos A. Catania[a], Facundo Bromberg[b], Carlos García Garino[a,c]

[a] *ITIC, Universidad Nacional de Cuyo, Mendoza, Argentina*
[b] *Dept. Sistemas de Información, FRM - UTN, Mendoza, Argentina*
[c] *Facultad de Ingeniería, Universidad Nacional de Cuyo, Mendoza, Argentina*

Abstract

In the past years, several support vector machines (SVM) novelty detection approaches have been applied on the network intrusion detection field. The main advantage of these approaches is that they can characterize normal traffic even when trained with datasets containing not only normal traffic but also a number of attacks. Unfortunately, these algorithms seem to be accurate only when the normal traffic vastly outnumbers the number of attacks present in the dataset. A situation which can not be always hold

This work presents an approach for autonomous labeling of normal traffic as a way of dealing with situations where class distribution does not present the imbalance required for SVM algorithms. In this case, the autonomous labeling process is made by SNORT, a misuse-based intrusion detection system. Experiments conducted on the 1998 DARPA dataset show that the use of the proposed autonomous labeling approach not only outperforms existing SVM alternatives but also, under some attack distributions, obtains improvements over SNORT itself.

Keywords: Anomaly detection - Intrusion Detection Systems - SVM - Labeling

1. Introduction

In the past years network security has become a serious problem. In the early years of the Internet, the set of network protocols that supported it worked reasonable well. However as the Internet grew, underlying security

faults in those protocols were observed. Security faults in protocols such as ARP, TCP, TELNET, SMTP and FTP have caused most of known attacks against network data confidentiality, authenticity and availability. Currently most of these problems have been fixed, however new ways to develop attacks are discovered everyday.

Network managers must be well prepared in order to prevent network attacks, e.g., being informed about new vulnerabilities. For several years, intrusion detection systems (**IDS**) provided an invaluable help to network managers, becoming an integral part of any network security package.

In the intrusion detection field two different approaches can be observed: misuse detection and anomaly detection (Mukherjee et al., 1994). The main idea behind misuse detection is to represent attacks in a form of a pattern or a signature in such a way that even variations of these attacks can be detected. Based on these signatures, this approach detects attacks through a large set of rules describing every known attack (Tsai et al., 2009; Wu & Yen, 2009). The main disadvantage of the signature based approach is its difficulty for detecting unknown attacks. The main goal of the anomaly detection approach is to build a statistical model for describing normal traffic. Then, any deviation from this model can be considered an anomaly, and recognized as an attack. Notice that when this approach is used, it is theoretically possible to detect unknown attacks, although in some cases, this approach can lead to a high false attack rate. This ability to detect unknown attacks has been the cause of the increasing interest in developing new techniques to build models based on normal traffic behavior in the past years.

The anomaly detection approach has been a very active research topic inside the machine learning community and it has been the subject of many articles over the past years. One of the most successful approaches is based on the idea of collecting data only from network normal operation. Then, based on this data describing normality, any deviation would be considered an anomaly. Different techniques were proposed for characterizing the concept of normality (Lee & Stolfo, 1998; Hofmeyr et al., 1998; Catania & García Garino, 2008). In practice, however, it is difficult to obtain clean data to implement these approaches. Verifying that no attacks are present in the training data can be an extremely hard task, and for large samples this is simply infeasible. On the other hand, if the data containing attacks is assumed attack free, intrusions similar to the ones present in the training data will be accepted as normal patterns, resulting in inaccurate models and consequently, an increment in the number of misdetections.

Recently, different authors proposed the use of unsupervised algorithms for dealing with datasets presenting not only normal traffic but also a considerable number of attacks (Eskin et al., 2002; Feng et al., 2005; Laskov et al., 2004). This situation could be considered more suitable than using datasets with only normal traffic instances. In this sense, SVM for novelty detection (Tax & Duin, 1999; Schölkopf et al., 2001) was proposed as an alternative approach with a significant success rate.

Unfortunately, as noticed by Eskin et al. (2002), SVM for novelty detection works under the assumption that the number of normal traffic instances vastly outnumbers the number of anomalies. Eskin suggests datasets with a proportion of at least 98.5% of normal traffic.

To the best of the authors knowledge, there is no study which confirms the number of attacks laying under such low proportion. Informal observations of real traffic however, show that it is possible to find periods of time where the number of attacks presents in traffic could easily outnumber normal traffic instances. This situation can be observed in commonly used datasets for intrusion detection evaluation such as the 1998 DARPA dataset (Lippmann et al., 2000). This dataset was provided by DARPA to the machine learning community in the context of the 1999 KDD Cup for evaluating different IDS approaches. Since its publication DARPA dataset has been widely used by many IDS researchers over the years. Interestingly, the 1998 DARPA class distribution does not exhibit the required imbalance. Moreover, the percentage of attacks present in the dataset is around 50%. Certainly, under these situations algorithms such as SVM for novelty detection could suffer considerable performance loss.

To deal with these imbalanced class distribution situations a novel approach is proposed. The idea is to provide a strategy for autonomous labeling only normal traffic, following the hypothesis that using an autonomous labeling tool may help reducing the presence of attacks in the traffic instances used for training, and consequently improving the performance of SVM for novelty detection. In this work, SNORT (Roesch, 1999), a very well known misuse signature-based IDS system, is proposed as a strategy for autonomous labeling normal traffic.

The rest of the work is organized as follows: in Section 2 main characteristics of SVM for novelty detection are briefly discussed, together with its application to the traffic network detection field. Then, in Section 3, a new approach for autonomous labeling normal traffic is presented. In section 4 a set of experiments is conducted on the 1998 DARPA dataset in order to

evaluate the performance of the different approaches. Finally, conclusions and future work are provided in Section 5.

2. SVM for novelty detection

Since its introduction in the mid-1990s (Boser et al., 1992; Cortes & Vapnik, 1995; Vapnik, 1998), The SVM algorithm has been widely used, being the subject of many articles on classification and other pattern recognition problems (Lee & Verri, 2002).

SVM approach for classification differ from other classification algorithms by three important properties. First, its formulation presents an important theoretical result, proving that the generalization error is minimized when the *margin* is maximized, where the margin is defined as the distance of the solution hyperplane to its closest point (Vapnik, 1998). This property is unique to SVM and is one of its main advantages when compared to other classification algorithms. Another important property is that the search for the maximal margin is a convex (quadratic) optimization problem, i.e., with only one minima, resulting in an efficient learning stage. In most cases, the input data points are not separable by the separation surface, so a standard approach (first introduced for the Perceptron algorithm of Rosenblatt (1958)), is to project the data points to higher dimension *feature* space. That usually affects the generalization error. However, for SVM, it can be proven (Vapnik, 1998) that for the maximal margin, the generalization error is still minimal, regardless of the dimension of the feature space. Finally, the formulation of the optimization problem (as shown in the next section for SVM for novelty detection) can be expressed solely in terms of the dot product between the feature vectors (denoted its *kernel*), which further reduces the computational complexity by permiting an efficient pre-computation of these quantities.

SVM for novelty detection is a generalization of the core SVM ideas for classification problems. Traditional SVM approaches for classification uses as input training data consisting of a mixture of data labeled by two classes. In the intrusion detection problem this would consist of data labeled both as *attack* and *non-attack*. The model constructed by these approaches discriminates the input space in two infinite regions, one per class, using a hyperplane as a separation surface. In contrast, the main idea in SVM for novelty detection (Tax & Duin, 1999; Schölkopf et al., 2001) is to use as input a description of only the *normal* class of objects (*non-attack* in IDS), assuming the rest

as *anomalies* (in our problem, the *attacks*). The model constructed by this approach discriminates the input space in a finite region containing the normal objects, while all the rest of the (infinite) space is assumed to contain the anomalies.

The SVM for novelty detection variants appear in the literature of intrusion detection with different names, which could lead to some confusion. In some cases they are referenced as SVM one-class algorithms. SVM for non supervised learning is another widely used name by some authors. Although, all of these names describe important characteristics of this kind of algorithms, in this work the term SVM for novelty detection will be preferred.

Two major approaches were proposed for generalizing SVM to the problem of novelty detection. One approach, proposed by Tax & Duin (1999), is based on the idea of finding a hypersphere with center $\mathbf{c}$ and minimal radius R containing the *normal* data, discriminating all other data not in the sphere as *anomalies*. As in standard SVM approaches, the discriminating surface (the sphere), as well as the data, may be mapped into a higher dimension feature space by a kernel function (see more details in next section). Another approach proposed by Schölkopf et al. (2001) tries to separate the normal data points from the anomalies by finding the hyperplane that is maximally distant from the origin. When a RBF kernel is used, it was shown that the two approaches converge to the same solution (Campbell, 2000). In this work the Tax's approach is preferred, which is explained in more detail below. For a description of Schölkopf's hyperplane formulation the reader is referred to Schölkopf et al. (2001).

2.1. SVM based on the hypersphere formulation

The sphere formulation has an intuitive geometric idea: the normal data $\{\mathbf{x}_i, i = 1, \ldots, N\}$ can be concisely described by a sphere, of center $\mathbf{c}$ and radius R, first projecting the data to some high-dimensional feature space by the mapping Φ, obtaining the projected set of points $\{\Phi(\mathbf{x_i}), i = 1, \ldots, N\}$, and assuming the projected *normal* points lie within the sphere. A graphical example of this can be observed in Fig. 1. Non-separability of the training data in the feature space can be addressed by introducing slack variables $\{\xi_i, i = 1, \ldots, N\}$, one per data point $\mathbf{x}_i$. The use of slack variables allows for some (projected) normal data points to lay outside the sphere. Although, this may lead to a number of (projected) anomalies lying within the sphere as well.

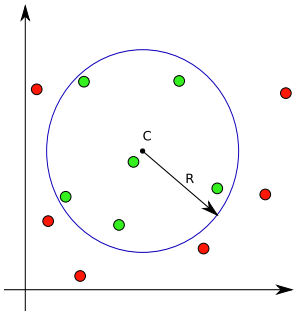


Figure 1: The geometric representation of the sphere formulation. The sphere of center $\mathbf{c}$ and the minimal radius R which enclose all the normal data points

The main insight of the hypersphere formulation is to note that the margin maximization approach of standard SVM maps into the hypersphere formulation by finding, among all possible hyperspheres that encloses all the normal data points, the one with smaller volume. Formally,

$$\underset{R\in\Re,\xi\in\Re^N}{min} R^2 + \frac{1}{\nu N}\sum_{i=1}^{N}\xi_i \tag{1}$$

where besides of minimizing the radius R, it minimizes the size of the slack variables. The constant ν gives the trade-off between the two terms: volume of the sphere and the number of target objects rejected.

To enforce the fact that normal points, minus their non-negative slacks, lies within the sphere, the above minimization is subject to the following constraints:

$$\begin{aligned} \left(\Phi(\mathbf{x}_i)-\mathbf{c}\right)\left(\Phi(\mathbf{x}_i)-\mathbf{c}\right)^T &\leq R^2+\xi_i \\ \xi_i &\geq 0. \end{aligned} \tag{2}$$

for all $i = 1, \ldots, N$, where the l.h.s. of the first constraint is no more than the distance of the feature vector $\Phi(\mathbf{x}_i)$ to the center $\mathbf{c}$ of the sphere.

To solve the constraint optimization problem (1) subject to constraints (2), the Lagrangian is minimized

$$
\begin{aligned}
&L(R, \mathbf{c}, \xi_i, \alpha_i, \beta_i) = \\
&\quad R^2 + \frac{1}{\nu N} \sum_i \xi_i - \\
&\quad \sum_i \alpha_i \left(R^2 + \xi_i - (\Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x}_i) - 2\mathbf{c} \cdot \Phi(\mathbf{x}_i) + \mathbf{c} \cdot \mathbf{c}) \right) - \\
&\quad \sum_i \beta_i \xi_i
\end{aligned} \tag{3}
$$

with Lagrange multipliers $\alpha_i \geq 0$ and $\beta_i \geq 0$. The standard trick in SVM that leads to a formulation based on kernels consists on minimizing the Lagrangian on all but the the Lagrange multipliers α_i. That is, the parcial derivatives w.r.t. R, $\mathbf{c}$, ξ_i are set to zero, to obtain the new constraints:

$$
\begin{aligned}
\sum_{i=1}^{N} \alpha_i &= 1, \\
\mathbf{c} &= \frac{\sum_i^N \alpha_i \Phi(\mathbf{x}_i)}{\sum_i^N \alpha_i} = \sum_i^N \alpha_i \Phi(\mathbf{x}_i) \\
0 &\leq \alpha = \left(\frac{1}{\nu N} - \beta_i \right) \leq \frac{1}{\nu N},
\end{aligned} \tag{4}
$$

for which, after resubstituting in the Lagrangian (3), a new Lagrangian over α_i is obtained

$$
\begin{aligned}
L(\alpha_i) &= \sum_i^N \alpha_i \left(\Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x}_i) \right) - \sum_{i,j}^N \alpha_i \alpha_j \left(\Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x}_j) \right) \\
&= \sum_i^N \alpha_i k(\mathbf{x}_i, \mathbf{x}_i) - \sum_{i,j}^N \alpha_i \alpha_j k(\mathbf{x}_i, \mathbf{x}_j).
\end{aligned}
$$

In the above equation one can see that the mapping $\Phi(\mathbf{x}_i)$ of datapoints $\mathbf{x}_i$ to a high-dimensional feature space can be formulated solely by the kernel function $k(\mathbf{x}_i, \mathbf{x}_j)$, a function over the inner product $\Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x}_j)$ in the feature space. Commonly used kernels are linear, sigmoid, polynomial, among others.

One of the most successful kernels used in the field of network traffic anomaly detection is the radial basis function (RBF), shown in Eq.(5)

$$k(\mathbf{x}_i, \mathbf{x}_j) = e^{-\gamma(\Phi(\mathbf{x}_i)-\Phi(\mathbf{x}_j))^2}, \tag{5}$$

where $\gamma = \frac{1}{\sigma^2}$. Notice that σ indicates the width, or spread, of the kernel function.

Finally, the learned model is used to classify between normal and anomalous traffic simply by computing whether a new object $\mathbf{z}$ is within the sphere, i.e., its distance to the center of the sphere is smaller than the radius:

$$\begin{aligned}(\Phi(\mathbf{z}) - \mathbf{c})\,(\Phi(\mathbf{z}) - \mathbf{c})^T = \\ k(\mathbf{z}, \mathbf{z}) - 2\sum_i \alpha_i k(\mathbf{z}, \mathbf{x}_i) + \sum_{i,j} \alpha_i \alpha_j k(\mathbf{z}, \mathbf{x}_j) \leq R^2\end{aligned} \tag{6}$$

where $\mathbf{c}$ is equated with $\sum_i \alpha_i \Phi(\mathbf{x}_i)$ according to Eq.(4). To compute the above inequality, it is necessary to find a way to obtain the radius R. For that, first note that the above inequality corresponds with the first constraint in Eq. (2). The Lagrangian optimization theory states that for those objects for which the constraint is satisfied with an equality, the Lagrange multipliers satisfy $\alpha_i \neq 0$. Those objects are called the *support vectors*. To compute R then, Eq. (6) must be solved for any of these support vectors.

2.2. Previous work on SVM for novelty detection in intrusion detection

Different authors (Eskin et al., 2002; Li et al., 2003; Laskov et al., 2004) have used SVM for novelty detection in the intrusion detection field. The work of Eskin et al. (2002) is one of the first on the subject. They propose a geometrical framework to improve the performance of different kind of unsupervised learning algorithms among which SVM is found. Laskov et al. (2004) used the same geometrical framework presented by Eskin and they provide a modification to SVM for novelty detection which outperforms traditional variants. Both works use the KDD99 DARPA dataset for training and evaluating their approach.

The work of Li et al. (2003) proposes an improvement on SVM for novelty detection applied to the intrusion detection field. The idea is basically to extend hyperplane-to-origin approach of Schölkopf et al. (2001). In their article, they assume that not only the origin lies in the second class but also that all data points close enough to the origin are to be considered as

outliers or anomaly data points. For the evaluation process of their approach the authors use the 1999 DARPA dataset.

It seems clear that all these authors are aware of the limitations of the different SVM approaches for anomaly detection. As mentioned by Eskin, these algorithms will work reasonably well under the assumption that the number of normal traffic instances vastly outnumbers the number of anomalies. Moreover, in the experiments conducted by Eskin et al. (2002) they assume that a high imbalance in class distribution is a common feature in network traffic and they have altered the original data sets to fit into this assumption. Unfortunately in practice, this assumption is not always valid. There are many situations in which for specific periods of time, the presence of intrusions vastly exceeds the number of normal traffic instances. For instance, when a new vulnerability is discovered and it has been widely announced, it is possible to find attacks exploiting these vulnerability encompassing a extremely high percentage of the network traffic. Thus, it seems that anomalies in network traffic have a bursty behavior. This can be observed in the DARPA dataset, where the percentage of anomalous traffic found in some weeks is less than 0.5% but in some other weeks the percentage raises to 70%. However, this dataset may not be representative of the actual imbalance in a production environment. The authors are unaware of a thorough study that confirms these claims.

It seems clear that under real traffic situations it is not always possible to guarantee the required class distribution for training sets, as needed by SVM approaches. A possible solution is to rely on experts for removing known attacks from the training set, until the desired imbalance is reached. This, however, would be an extremely expensive and tedious task. Perhaps, a more appealing idea consist of using an autonomous labeling tool for removing a considerable number of well-known attacks.

3. Proposed approach: Autonomous labeling of normal traffic using SNORT.

An autonomous labeling approach is proposed for dealing with non-imbalanced class distributions, The idea behind this approach is to provide mechanisms for excluding well-known attacks from the dataset. Well-known attacks are the ones whose behavior have been deeply analyzed and a set of rules haven been built for describing such behavior. There are many tools for detecting well-known attacks. In particular, recognizing well-known attacks is a

common task done by traditional signature-based IDS. Thus, the use of IDS as an autonomous labeling tool can provide a good mechanism for reducing the number of attacks in the training dataset required for SVM for novelty detection algorithm.

The complete process is graphically represented in Figure 2. Given a dataset containing unlabeled network traffic instances, an autonomous labeling tool is used for labeling the dataset. Then, traffic instances labeled as attack are discarded whereas the remaining ones are labeled as presumably normal and used for training SVM for novelty detection.

Notice that there is no guarantee the labeling process could be done without errors. However, the assumption is that after the attacks recognized by the autonomous labeling tool are removed from the training data set, the number of normal traffic instances will be sufficiently larger than the number of attacks. This way, class distribution becomes unbalanced or at least closer to the suggested imbalance.

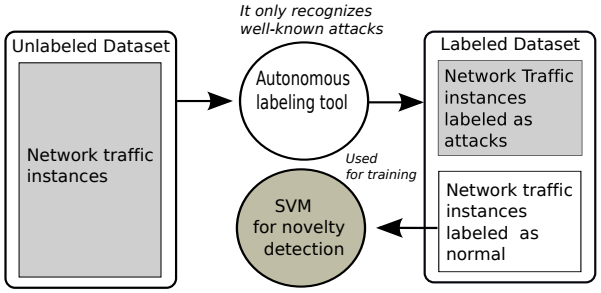


Figure 2: Training process using an autonomous labeling tool on a dataset

The autonomous labeling tool proposed in this work is SNORT (Roesch, 1999), a light and fast intrusion detection system developed by Martin Roesch in 1999. Over the past years, its popularity grew considerably, becoming a de-facto standard in the security network field. SNORT is composed by several fast pattern matching algorithms and a very complete and updated rule database. In recent versions, SNORT has included preprocessors for flow tracking and IP defragmentation which has improved its overall detection performance.

However, SNORT is far from being a complete solution to the intrusion problem. As any other misuse signature-based IDS, SNORT fails to recognize many attacks which are not described by a rule of its database. Another well known problem is that in many cases, SNORT can raise an extremely high

false alarm rate, leading to production of different approaches for reducing SNORT false alarm (Tjhai et al., 2008).

The main hypothesis of this work is that although SNORT may present a considerable number of misclassifications, it still can be useful for reducing the proportion of attacks in the dataset and consequently producing potentially better results for SVM for novelty detection.

4. Experiments

This section evaluates the performance of the SVM for novelty detection when SNORT is used as an autonomous labeling tool. A set of experiments are conducted comparing performance of the proposed approach (denoted here as *SbSVM*) against the standard SVM algorithm for novelty detection. A comparison against Standalone SNORT performance is also conducted in order to establish in which situations *SbSVM* performance is below performance shown by standalone SNORT. In those situation, the use of SNORT by itself will be more convenient and the *SbSVM* approach should be avoided.

4.1. Dataset description

The experiments were conducted over five weeks of the 1998 DARPA data set (Lippmann et al., 2000), widely used for intrusion detection evaluation. DARPA dataset contains around 1.5 millions traffic instances with almost 50% of them labeled as attacks.

For describing the input data, a total of six fields from a network traffic instance were selected: connection time, protocol type, source port, destination port, source IP address and destination IP address. These fields have been used in previous works (Catania & García Garino, 2008) and have provided a good trade off between overall performance and the computational effort needed for training process.

Selected fields are represented according to Table 1 resulting in a total of 14 attributes used for training SVM for novelty detection alternatives.

To improve SVM performance and to avoid possible numerical problems, features are normalized between the interval [0,1] as suggested in Hsu et al. (2008).

4.2. Dataset sampling

A randomly selected 1% subset of the DARPA data is used for the training process, whereas another 0.5% subset is used for testing purposes, following standard ratios used in classification problems.

Table 1: Features representation

Feature	Size
Connection time	3
Protocol Type	1
Source port	1
Destination port	1
Source IP address	4
Destination IP address	4

For the training process, standard SVM approach uses the whole 1% including both normal and anomalous traffic. In the case of the *SbSVM* approach, as mentioned in section 3, attacks recognized by SNORT are removed, resulting in a ratio smaller than 1%. On the other hand, as SNORT does not require a training process, only an evaluation process is carried out against the 0.5% subset.

In order to evaluate the influence different attack distributions have on classifiers performance, experiments are conducted against datasets containing distributions with 1.0% 2.0%, 5.0%, 10%, 20%, 50%, 60% and 80% of attacks. The 0.01 fraction of the whole DARPA dataset (i.e., 1% of it) with a proportion of attacks p% is sampled from the whole dataset in two steps, one that samples attacks from the set of all attacks, and another for sampling the normal data from the set of all normal traffic instances. To maintain the p% ratio of attacks in the resulting 1% dataset, a fraction $p \times 10^{-4}$ of attacks are randomly and uniformly sampled from the set of all attacks. Similarly, a fraction of $(1-p) \times 10^{-4}$ is randomly and uniformly sampled from the set of all normal traffic instances.

For statistical significance a total of 20 repetitions of the experiments are conducted using different randomly and uniformly selected subsets for each attack distribution.

4.3. Performance metrics for IDS evaluation

Standard performance metrics for IDS evaluation are used for comparing the different approaches discussed. These metrics correspond to Attack Detection rate (DR) and False Alarm rate (FA).

DR is computed as the ratio between the number of correctly detected attacks and the total number of attacks. Whereas FA rate is computed as

the ratio between the number of normal connections that are incorrectly classified as attacks and the total number of normal connections.

4.4. Standalone SNORT evaluation

Before evaluating the proposed labeling approach, it is important to evaluate the classification performance (in normal traffic and attacks) of standalone SNORT. Notice that the version of SNORT used in this experiment is 2.8.3.2.

From a total of thousands of rules in the SNORT rule-base, only 32 matched against the whole 5 weeks of the DARPA data set. Therefore, for improving further computations the unmatching rules were removed from SNORT's rule-base. The complete rule-base is shown in Table A.5.

Table 2 shows averaged results for DR and FA, as well as their respective standard deviations (sd) over the 20 repetitions. It can be observed that averaged results for FA and DR obtained by SNORT do not present a significant variation as attack distribution grows. These results can be expected because SNORT uses the same set of rules over all of the attack distribution datasets. The performance presented by SNORT on DR for each attack distribution is around 87% and in the case of FA, the obtained value vary slightly around 4%.

Table 2: SNORT performance evaluation on DARPA data set with different attack distributions

Attack Distribution (%)	DR (%)	sd	FA (%)	sd
1	87.15	4.09	4.45	0.21
2	86.59	3.03	4.43	0.28
5	87.13	1.71	4.37	0.20
10	86.67	1.18	4.41	0.24
20	86.88	1.07	4.35	0.31
50	86.84	0.59	4.38	0.27
60	87.00	0.45	4.47	0.40
80	87.02	0.43	4.52	0.47

Despite a number of misclasifcations, SNORT shows a very accurate performance on attack detection rate for the DARPA dataset. Therefore, it is expected that *SbSVM* can bring class distribution closer to the imbalance required by SVM algorithms for novelty detection.

4.5. Evaluation of the SNORT-based autonomous labeling for SVM novelty detection

The SVM implementation used in these experiments is an extension of the *libsvm* (Chang & Lin, 2001) that supports the hypersphere formulation (Russo, 2008).

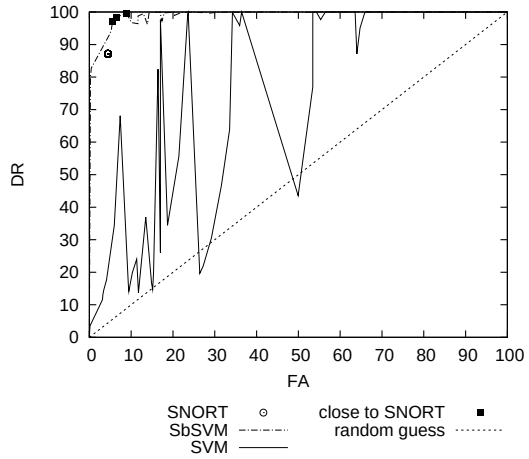
An RBF kernel is chosen for both approaches (*SbSVM* and standard SVM). The use of an RBF kernel implies finding an appropriate value for γ (see Eq. (5)). Therefore, the performance of *SbSVM* and the standard SVM algorithm is evaluated for different combinations of γ and the constant ν (see Eq. (1)). Selected values for γ and ν are shown in Table 3. Although the selected values are far from a complete parameter set, they include a considerable parameter range, suitable for evaluating the proposed approach. Results provided by *SbSVM* are compared with the ones computed using standard SVM for novelty detection, as well as the ones provided by the standalone SNORT classifier.

Table 3: Selected values for γ and ν

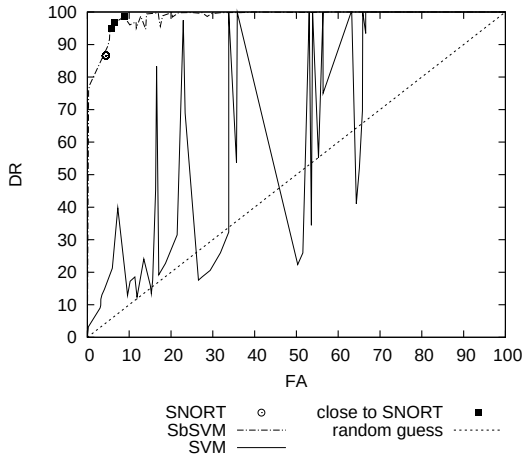
ν	0.0	0.01	0.1	0.2	0.27	0.4	0.6	0.7	0.9	1.0
γ	1	2	4	8	12	20	35			

FA and DR values for each γ and ν combination are used for generating ROC curves (Fawcett, 2006) for both approaches. Figure 3 and Figure 4 show ROC curves together with results from SNORT standalone classifier. In the case of standalone SNORT however, evaluation results are plotted only as one dot on the figures (as it is independent of γ and ν) whereas those γ and ν combinations which performance is close to standalone SNORT are also plotted on the ROC curves with filled squares.

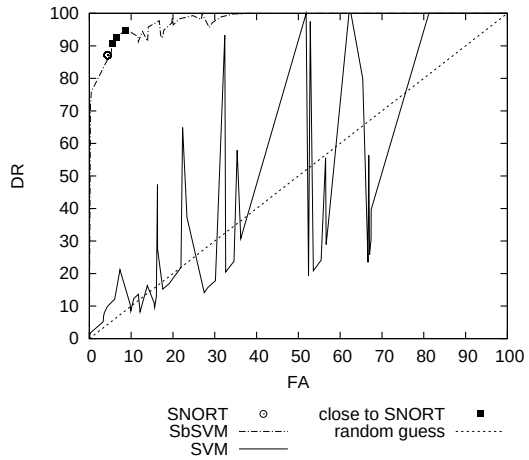
Figure 3 shows that under 1%, 2%, 5% and 10% attack distributions, performance exhibited by *SbSVM* clearly outperforms standard SVM results. On the one hand, for dataset containing 1% and 2% of attacks, standard SVM presents most of the results above random guess line. However, this behavior changes for the remaining datasets, where most of the results remain under guess line. Instead, the proposed *SbSVM* approach does not suffer from this behaviour and maintains all its results above random guess line. On the other hand, classification performance shown by standard SVM clearly decreases as attack distribution grows, whereas in the case of *SbSVM*, only a slightly performance loss can be appreciated as attack number grows up to 10%.



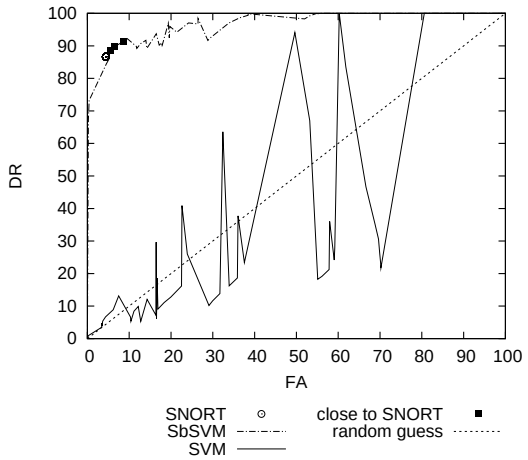
(a) 1% attack distribution



(b) 2% attack distribution



(c) 5% attack distribution



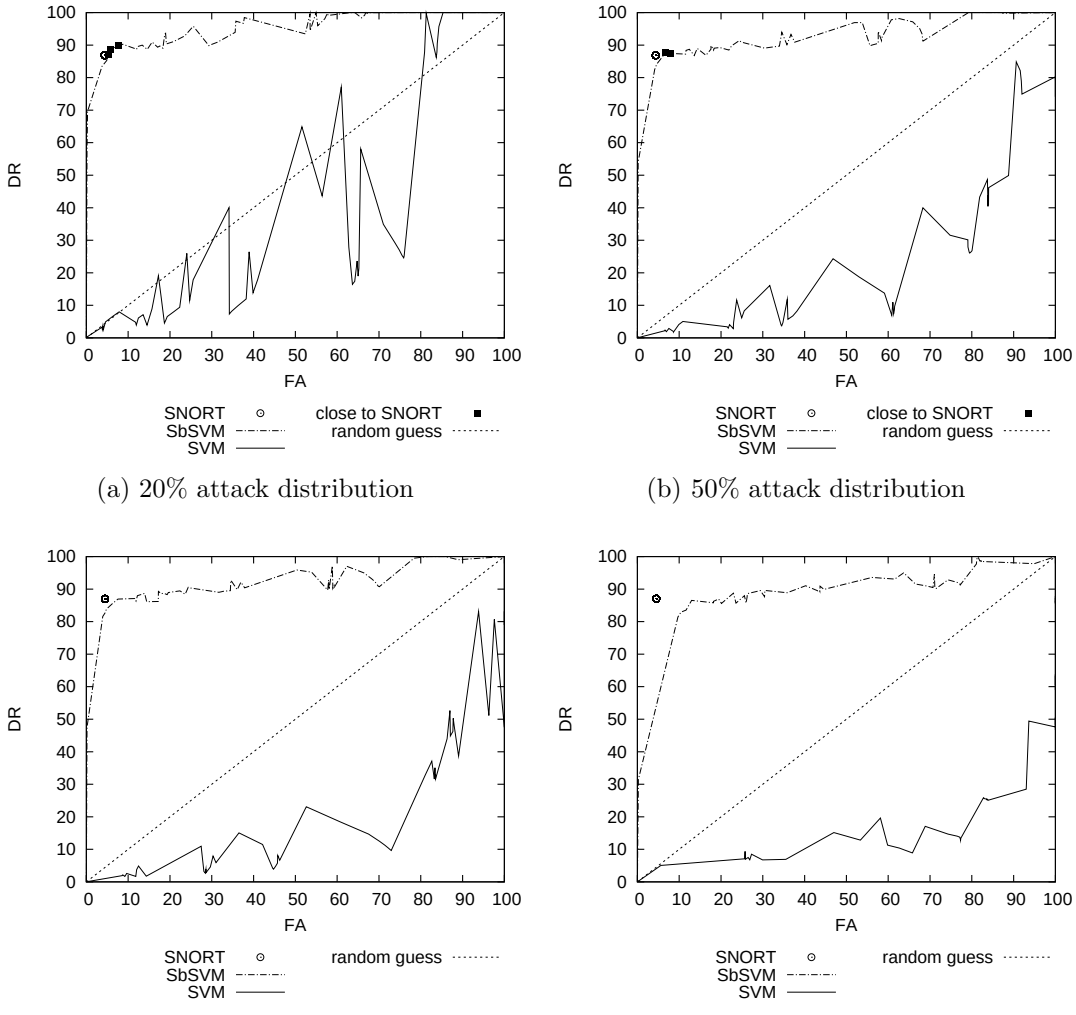
(d) 10% attack distribution

Figure 3: ROC curves under 1%, 2%, 5% and 10% attack distribution

Figure 4 shows remaining ROC curves for datasets under 20%, 50%, 60% and 80% attack distribution. Despite the appreciable performance loss for datasets with attack distributions above the 20%, the proposed *SbSVM* approach continues outperforming standard SVM. In this sense, it can be observed that for datasets under 20% attack distribution, standard SVM presents only two parameter combination above random guess line, whereas

for the remaining dataset distributions, ROC curves for standard SVM decrease far below random guess line. In contrast, the proposed *SbSVM* maintains all its results above random guess line.

Another noticeable disadvantage observed by the standard SVM algorithm is that all the ROC curves show a variable behaviour with abrupt performance changes along different parameters combination. A situation which is not exhibited by *SbSVM*.



(a) 20% attack distribution

(b) 50% attack distribution

(c) 60% attack distribution

(d) 80% attack distribution

Figure 4: ROC under 20%, 50%, 60% and 80%

Table 4 shows detailed performance information about the three parameter combinations that present the best trade off between DR and FA for each approach on every attack distributions. In other words, the highest values for DR while keeping FA values as low as possible.

DR values higher than 87% together with a FA values lower than 10% are considered suitable for real traffic situations (values very close to the performance showed by SNORT in Section 4.4). Those parameter combinations whose performance is close to these values are highlighted on the table. (It is worth to notice that these values correspond to the filled squares plotted in Figure 3 and Figure 4).

Table 4: Classification performance under different attack distribution

(a) *SbSVM*

%	ν	γ	DR %	sd	FA %	sd
1%	0.10	2.00	**99.65**	0.74	**8.84**	0.54
	0.10	4.00	**98.40**	1.41	**6.51**	0.42
	0.10	6.00	**97.08**	1.91	**5.66**	0.36
2%	0.10	2.00	**98.55**	1.09	**8.78**	0.42
	0.10	4.00	**96.69**	2.24	**6.50**	0.33
	0.10	6.00	**95.03**	2.54	**5.71**	0.26
5%	0.10	2.00	**94.81**	1.24	**8.62**	0.55
	0.10	4.00	**92.49**	1.37	**6.46**	0.36
	0.10	6.00	**90.84**	1.37	**5.63**	0.28
10%	0.10	2.00	**91.31**	1.19	**8.54**	0.32
	0.10	4.00	**89.68**	1.20	**6.41**	0.39
	0.10	6.00	**88.51**	1.38	**5.57**	0.37
20%	0.10	2.00	**89.80**	0.81	**7.73**	0.34
	0.10	4.00	**88.76**	0.87	**5.83**	0.31
	0.10	6.00	**87.01**	1.05	**5.18**	0.32
50%	0.10	1.00	**87.43**	0.56	**7.90**	0.58
	0.10	2.00	**87.64**	0.83	**6.73**	0.61
	0.20	8.00	**88.38**	0.63	11.63	0.69
60%	0.20	4.00	**88.67**	0.33	13.94	0.81
	0.20	6.00	**87.98**	0.47	12.22	0.70
	0.20	8.00	**87.09**	0.44	11.56	0.74
80%	0.20	2.00	**88.76**	2.62	22.97	17.72
	0.27	6.00	**88.60**	2.68	26.31	16.95
	0.20	1.00	**88.55**	2.67	23.02	17.73

(b) SVM for novelty detection

%	ν	γ	DR %	sd	FA %	sd
1%	0.27	2.00	99.65	0.86	23.64	0.67
	0.27	1.00	98.96	1.15	23.51	0.63
	0.20	1.00	97.64	2.49	17.00	0.57
2%	0.40	2.00	99.86	0.28	35.86	0.77
	0.40	1.00	99.69	0.34	33.82	0.65
	0.27	1.00	97.48	1.12	22.94	0.55
5%	0.60	1.00	99.94	0.11	51.85	0.79
	0.60	2.00	97.49	2.22	52.85	0.49
	0.40	1.00	93.27	1.86	32.40	0.53
10%	0.70	1.00	99.96	0.08	60.26	0.60
	0.60	1.00	94.10	0.98	49.63	0.57
	0.70	2.00	83.55	2.23	61.82	0.61
20%	0.70	1.00	77.10	1.30	60.99	0.66
	0.60	1.00	64.89	1.59	51.57	0.62
	0.70	2.00	58.07	1.59	65.63	1.01
50%	0.60	1.00	39.99	0.86	68.31	0.88
	0.60	2.00	31.54	0.87	74.80	0.76
	0.60	35.00	30.14	1.13	79.06	0.70
60%	0.70	35.00	52.67	1.11	86.98	0.63
	0.70	20.00	50.28	1.21	87.76	0.59
	0.70	12.00	47.92	1.20	88.00	0.55
80%	0.40	2.00	25.86	17.02	82.79	4.73
	0.40	4.00	25.54	17.09	83.10	3.99
	0.40	8.00	25.47	17.11	83.63	3.86

In the case of the dataset under 1% attack distribution, standard SVM provides barely good enough results. For the first two parameter combinations, a near-optimal detection rate is obtained. Unfortunately together with FA values around 23%, which are considered excessively high for practical uses. More appropriate are the results provided by the third combination,

where DR value remains high (97%) and FA value decreases to 17%. Beyond this attack distribution, standard SVM presents a significant performance loss. Moreover, in all these remaining attack distributions, none of the results obtained by SVM exhibit a classification performance suitable for real traffic situations.

On the other hand, the *SbSVM* approach provides considerable better results. From dataset from 1% to 50% *SbSVM* shows DR values from 87% to 99% with FA values oscillating from 5.5% to 8.8%, which as was mentioned in previous paragraphs, can be considered suitable for practical uses. For datasets under 60% attack distribution, DR values remain around 88%, however FA values oscillate around 12%. Degradation of FA values is even more noticeable on datasets under 80% attack distribution where FA values raise to an useless 26%.

Despite the fact that comparing the proposed approach against SNORT classifier is not the main focus of this work, it is worth to notice that many of the parameter combinations for the *SbSVM* approach shown in Table 4 present improvements over standalone SNORT classification performance, demonstrating generalization of SVM over SNORT's classifcation. *SbSVM* shows better-than-SNORT results for datasets with attack distributions up to 10%, where DR values oscillate from 88% to 97% at the expense of a slight performance loss on FA (around 5%).

On the other hand standard SVM could not provide a classification performance close the one exhibited by SNORT under any attack distribution.

5. Conclusions

Experiments showed that the overall performance of SVM based on the hypersphere formulation on the 1998 DARPA dataset decreases to unpractical values for datasets with more than 2% of attacks. These results seem to confirm what has been already discussed in Section 2.2 and references then on. When a high number of attacks are included in the dataset, SVM algorithms for novelty detection are not suitable for finding an accurate domain description. Thus, a highly imbalanced class distribution is needed in the dataset to achieve a proper performance.

The use of an autonomous labeling tool appears to be a promising strategy for handling classes without the required distribution. The proposed *SbSVM* approach provides significant better results than standard SVM. Major benefits are shown beyond 2% attack distribution, where standard SVM FA

shows values between 50% and 80% whereas *SbSVM* maintains values around 28%. Moreover, in the case of DR, *SbSVM* shows an improvement from two to eight times compared with the ones obtained with standard SVM along different attack distribution datasets.

The proposed *SbSVM* approach appears to be more robust along not only different attack distributions but also different parameter combinations. *SbSVM* maintains all of its results above random guess line, whereas standard SVM shows a variable behaviour with abrupt performance changes along over all attack distribution datasets.

The obtained results have shown that for datasets containing up to 50% of attacks, the autonomous labeling approach using SNORT has improved not only SVM algorithms for novelty detection but also standalone SNORT. For dataset with less than 20% attack distribution, *SbSVM* has improved more than 10% DR value while FA value has increased around 1%. Smaller but appreciable improvements have also been shown on dataset between 20% and 50%, where *SbSVM* has outperformed by a 2% SNORT DR. Beyond this point, using standalone SNORT seems to be the more convenient strategy.

The performance of the *SbSVM* approach in real traffic situations still remains unknown. Consequently, experiments will be carried out to overcome this issue in the future.

References

Boser, B. E., Guyon, I. M., & Vapnik, V. N. (1992). A training algorithm for optimal margin classifiers. In *Proceedings of the 5th Annual ACM Workshop on Computational Learning Theory* (pp. 144–152). ACM Press.

Campbell, C. (2000). Kernel methods: A survey of current techniques. *Neurocomputing*, *48*, 63–84.

Catania, C., & García Garino, C. (2008). Reconocimiento de patrones en el tráfico de red basado en algoritmos genéticos. *Inteligencia Artificial, Revista Iberoamericana de IA*, *12*, 65–75.

Chang, C.-C., & Lin, C.-J. (2001). *LIBSVM: a library for support vector machines*. `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

Cortes, C., & Vapnik, V. (1995). Support vector networks. *Machine Learning*, *20*, 273–297.

Eskin, E., Arnold, A., Prerau, M., Portnoy, L., & Stolfo, S. (2002). A geometric framework for unsupervised anomaly detection: Detecting intrusions in unlabeled data. In *Applications of Data Mining in Computer Security*. Kluwer.

Fawcett, T. (2006). An introduction to ROC analysis. *Pattern Recognition Letters*, *27*, 861–874.

Feng, Y., Wu, Z.-F., Wu, K.-G., Xiong, Z.-Y., & Zhou, Y. (2005). An unsupervised anomaly intrusion detection algorithm based on swarm intelligence. In *Machine Learning and Cybernetics. Proceedings of 2005 International Conference on* (pp. 3965–3969). volume 7.

Hofmeyr, S. A., Forrest, S., & Somayaji, A. (1998). Intrusion detection using sequences of system calls. *Journal of Computer Security*, *6*, 151–180.

Hsu, C.-W., Chang, C.-C., & Lin, C.-J. (2008). A practical guide to support vector classification. http://www.csie.ntu.edu.tw/ cjlin/papers/guide/guide.pdf.

Laskov, P., Schafer, C., & Kotenko, I. (2004). Intrusion detection in unlabeled data with quarter-sphere support vector machines. In *Procedings of 2004 DIMVA* (pp. 71–82).

Lee, S.-W., & Verri, A. (Eds.) (2002). *Pattern Recognition with Support Vector Machines* volume 2388 of *Lectures Notes in Computer Science*. Springer Berlin.

Lee, W., & Stolfo, S. J. (1998). Data mining approaches for intrusion detection. In *Proceedings of the 7th USENIX Security Symposium*.

Li, K.-L., Huang, H.-K., Tian, S.-F., & Xu, W. (2003). Improving one-class svm for anomaly detection. In *Machine Learning and Cybernetics, 2003 International Conference on* (pp. 3077–3081). volume 5.

Lippmann, R., Fried, D., Graf, I., Haines, J., Kendall, K., McClung, D., Weber, D., Webster, S., Wyschogrod, D., Cunningham, R., & Zissman, M. (2000). Evaluating intrusion detection systems: the 1998 darpa offline intrusion detection evaluation. In *DARPA Information Survivability Conference and Exposition, 2000. DISCEX '00. Proceedings* (pp. 12 –26 vol.2). volume 2.

Mukherjee, B., Heberline, L. T., & Levitt, K. (1994). Network instruction detection. *IEEE Network*, *8*, 26–41.

Roesch, M. (1999). Snort - lightweight intrusion detection for networks. In *LISA '99: Proceedings of the 13th USENIX conference on System administration* (pp. 229–238). Berkeley, CA, USA: USENIX Association.

Rosenblatt, F. (1958). The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, *65*, 386–408.

Russo, V. (2008). LIBSVM Plus. `http://neminis.org/software/libsvm-plus/`.

Schölkopf, B., Platt, J. C., Shawe-taylor, J., Smola, A. J., & Williamson, R. C. (2001). Estimating the support of a high-dimensional distribution. *Neural Computation*, *13*, 1443–1471.

Tax, D., & Duin, R. (1999). Data domain description using support vectors. In *Proceedings of the European Symposium on Artificial Neural Networks* (pp. 251–256).

Tjhai, G. C., Papadaki, M., Furnell, S. M., & Clarke, N. L. (2008). The problem of false alarms: Evaluation with Snort and DARPA 1999 dataset. In *Trust, Privacy and Security in Digital Business* (pp. 139–150). Springer Berlin / Heidelberg volume 5185 of *Lecture Notes in Computer Science*.

Tsai, C.-F., Hsu, Y.-F., Lin, C.-Y., & Lin, W.-Y. (2009). Intrusion detection by machine learning: A review. *Expert Systems with Applications*, *36*, 11994 – 12000.

Vapnik, V. (1998). *Statistical learning theory*. Wiley Interscience.

Wu, S.-Y., & Yen, E. (2009). Data mining-based intrusion detectors. *Expert Systems with Applications*, *36*, 5605–5612.

Appendix A. Snort Complete Rule Base

Table A.5: Snort Rule set matched against 5 weeks of 1998 DARPA dataset

SID	Rule Description
[1:1156:10]	WEB-MISC apache directory disclosure attempt
[1:1418:13]	SNMP request tcp
[1:1419:12]	SNMP request tcp
[1:1420:13]	SNMP request tcp
[1:1421:13]	SNMP AgentX/tcp request
[1:1445:5]	POLICY FTP file_id.diz access possible warez site
[1:1762:8]	WEB-CGI phf arbitrary command execution attempt
[1:1842:20]	IMAP login buffer overflow attempt
[122:1:0]	TCP Portscan
[122:5:0]	UDP Portscan
[1:269:11]	DOS Land attack
[1:270:9]	DOS Teardrop attack
[1:3151:4]	FINGER / execution attempt
[1:323:6]	FINGER root query
[1:3274:7]	TELNET login buffer non-evasive overflow attempt
[1:330:10]	FINGER redirection attempt
[1:332:9]	FINGER 0 query
[1:335:6]	FTP .rhosts
[1:359:6]	FTP satan scan
[1:469:4]	ICMP PING NMAP
[1:491:8]	INFO FTP Bad login
[1:498:7]	ATTACK-RESPONSES id check returned root
[1:527:10]	BAD-TRAFFIC same SRC/DST
[1:546:6]	POLICY FTP 'CWD ' possible warez site
[1:547:6]	POLICY FTP 'MKD ' possible warez site
[1:584:13]	RPC portmap rusers request UDP
[1:598:13]	RPC portmap listing TCP 111
[1:612:7]	RPC rusers query UDP
[1:613:6]	SCAN myscan
[1:646:6]	SHELLCODE sparc NOOP
[1:716:15]	INFO TELNET access
[1:718:9]	INFO TELNET login incorrect