On Extreme Principles of Machine Learning in Anomaly and Vulnerability Assessment.

Conference Paper · January 2006

Source: DBLP

Dmitry Tunitsky

Russian Academy of Sciences

On Extreme Principles of Machine Learning in Anomaly and Vulnerability Assessment

Dr. Konstantin V. Malkov, Dr. Dmitry V. Tunitsky

Abstract— This paper concerns performance and accuracy limitations in the analysis of high-volume, high-dimension data within anomaly detection and analysis systems. We introduce an adaptive, machine-learning approach that ensures greater throughput (requires less computational resources) and progressively improved accuracy in the detection of and derivation of knowledge about atypical activity among very large data sets in dynamic computing (and other) environments.

***Index Terms*— adaptive systems, anomaly analysis, intrusion detection, kernel classifiers, machine learning.**

I. Introduction

EMERGING mathematical approaches designed to enable automated adaptivity of an anomaly analysis baseline have proven limited in their ability to deliver the accuracy, performance or reliability required of mission-critical applications [6], [7], [11]-[15]. This paper introduces an approach and algorithm designed to overcome specific and fundamental limitations in anomaly detection and analysis systems.

The design of this algorithm is based on three successive interrelated extreme principles.

1. The first of these three principles is a well known least squares method with specified weights, which enables the center of a training sample to be determined.
2. The second is a natural enhancement of the least squares method that enables the training sample center and related weights to be determined adaptively.
3. The third one – universal extreme principle detects the scaling factor for the decision rule.

Thus we describe an adaptive approach that identifies atypical events, calculates the extent of each event's deviation, and derives details regarding how the variables within the analytic model contributed to an event's deviation. The solution generates progressively improved output even when applied to complex models and very large data sets (i.e. gigabytes/terabytes) on standard Intel and SUN platforms. The solution has been applied to various types of security-related data (i.e. firewall, application, intrusion detection system, security information management, etc. Practical examples of the algorithm implementation are mentioned at the end of this article.

The approach, however, is applicable to a wide variety of challenges wherein the derivation of knowledge regarding atypical activity represents value, e.g. fraud detection, policy/regulatory compliance, equity/futures/currency trading, process optimization, marketing, homeland defense, etc.

II. Problem description

Consider an *input space* – a set Ω, which elements are called *events*, and its finite subset *X*, which is a *training sample*. The problem under consideration is to construct a learning machine that can assess how "typical" or "untypical" an event is from the input space Ω, with respect to the events of the training sample. To make it more precise, a *membership function*

$$w : \Omega \to [0,1]$$

on the input space should be defined in such a way that it has a greater value on more "typical" events with respect to the training sample, and a less value on less "typical". In other words, – a structure of a fuzzy set should be defined on Ω, i.e., the *membership degree* $w(x)$ for all the events $x \in \Omega$ should be specified [1, ch. 1.2].

Manuscript received January 28, 2006. This work was sponsored, supported, and funded by PWI, Inc. dba Privacyware.

Dr. Konstantin V. Malkov is a Director and Chief Technology Officer for PWI, Inc. Red Bank, NJ 07701 USA (732-212-8110 ext.236; fax: 732-212-9210; e-mail: kmalkov@privacyware.com). Ph.D. in Mathematics from Moscow State University, IEEE Member.

Dr. Dmitry V. Tunitsky is a Researcher at the Institute of Control Sciences, Russian Academy of Science, and Consultant to PWI, Inc. (732-212-8110 ext.236; fax: 732-212-9210; e-mail: dtunitsky@pwicorp.com). Ph.D. in Mathematics from Moscow State University.

III. Symmetric non-negative kernel

The described above problem is an unsupervised learning problem [2, ch. 1] since we don't have any expert assessment of membership degrees of the events from the training sample. We'll apply a well-known technical approach, which was worked out in the 1960th [3]. In contemporary literature this approach is called *kernel trick* [2, ch. 2.3], [4, ch. 3]. Namely, – create a mapping of the input space Ω to a Euclidian space *H* of a sufficiently high dimension:

$$\varphi : \Omega \to H .$$

The space *H* is called a *feature space*. For determinacy – let *H* be an infinite-dimensional Hilbert space. Thus events are mapped to the points of an infinite dimensional feature space.

The mapping φ itself has no particular importance. What is really important though is that this map induces a symmetric non-negative definite function

$$k : \Omega \times \Omega \to R^1 ,$$

that is called a *kernel* and is represented by the following formula

$$k(x, y) = \langle \varphi(x), \varphi(y) \rangle ,$$

where $\langle \cdot,\cdot \rangle$ is the inner product of the feature space *H*. Hence from the very start we can talk just about consideration for a symmetric non-negative definite kernel *k*. This kernel totally defines all the metric relations on the input space, i.e. we can measure both the distance $\rho(x, y)$ between any two events *x* and *y*:

$$\rho(x, y) = \sqrt{(\varphi(x) - \varphi(y))^2} = \sqrt{k(x,x) - 2k(x,y) + k(y,y)}$$

and the value of the angle $\angle(x, y)$ between them:

$$\angle(x, y) = \arccos\left(\frac{k(x,y)}{\sqrt{k(x,x)k(y,y)}} \right).$$

A particular choice of one or an other kernel is dictated by specific application domain where the problem in hand has arisen, and could vary significantly (see for example [2], [4], [5]). Below we will elaborate on this choice to be made, and consider a fixed symmetric non-negative definite kernel *k*.

IV. Least squares method

From geometrical point of view the following approach to solution of the problem under consideration looks quite natural:

- to find the "center" *c* of the training sample *X* in the feature space *H*,
- to estimate the membership degree of an event *x* as an inverse variation to the squared distance of it's image $\varphi(x) \in H$ to that "center".

Thus the decision rule will be

$$w(x) = \frac{a}{a + (\varphi(x) - c)^2} , \qquad (1)$$

where *a* is a fixed positive number, playing a role of a scaling coefficient. To be unambiguous, we'll consider the training sample with *N* events,

$$X = \{ x_i \}_{i=1}^{N} ,$$

where *N* is a positive integer number. Let us assume the weight coefficient w_i to be fixed for each event x_i of the training sample and use the least squares method for detecting of the "center" *c*. Namely, choose such a point

$$c \in H$$

as the "center" of the training sample that delivers minimum to the quadratic functional

$$J_0(c) = \frac{1}{2} \cdot \sum_{i=1}^{N} w_i^2 (\varphi(x_i) - c)^2 .$$

The value of this functional is a weighted sum of the squared distances of the event's images in feature space to the "center" *c*. Since

$$J_0'(c) = 2 \cdot \sum_{i=1}^{N} w_i^2 (\varphi(x_i) - c),$$

the functional $J_0(c)$ has the unique extremum at the point

$$c = \frac{\sum_{i=1}^{N} w_i^2 \varphi(x_i)}{\sum_{i=1}^{N} w_i^2} . \qquad (2)$$

V. ADAPTIVE CHOICE OF WEIGHT COEFFICIENTS

In the previous section it was described how to find explicitly a solution of the learning problem, set in section II by means of the least squares method. One disadvantage of the described method is a necessity of an a priori definition of the weight coefficient for each event of the training sample. Constructing an adaptive (with respect to the training sample) procedure of finding the weight coefficients can eliminate this disadvantage. To implement such a construction we will impose on each event x_i of the training sample an additional condition, based on a coincidence of each weight coefficient w_i with the membership degree of the corresponding event x_i, i=1,…, N, i.e.,

$$w_i = w(x_i)\,.$$

Therefore simultaneously for all i=1,…, N the conditions

$$w_i = \frac{a}{a + (\varphi(x) - c)^2}$$

must be met, where c is defined by expression (2). Reducing these conditions and expression (2) to common denominator, we obtain the following equations:

$$w_i\left(a + (\varphi(x) - c)^2\right) - a = 0\,,$$

$$\sum_{i=1}^{N} w_i^2 (\varphi(x_i) - c) = 0\,.$$

It's easy to see that the left parts of the obtained equalities coincide with partial derivatives $\frac{\partial J_1}{\partial w_i}(c, w_1,..., w_N)$ and $\frac{\partial J_1}{\partial c}(c, w_1,..., w_N)$ respectfully of the functional

$$J_1(c, w_1,..., w_N) = \frac{1}{2}\cdot\sum_{i=1}^{N}\left(w_i^2 (\varphi(x_i) - c)^2 + a(1 - w_i)^2\right)$$

(cmp. to [6]). Thus to solve the learning problem set in section II it's sufficient to find an extreme point $(c, w_1,..., w_N)$, that delivers minimum to the functional $J_1(c, w_1,..., w_N)$.

VI. ADAPTIVE CHOICE OF SCALING COEFFICIENT

The previous section describes how the extreme principle is used to solve the problem set in section II. Namely, – both the "center" c and the weight coefficients $w_1,..., w_N$ (of the training sample) were simultaneously detected. Subsequently – section V describes how to deal with one of the main disadvantages of the least squares method described in section IV. The mentioned disadvantage is caused by the necessity of an a priori definition of the weight coefficient for each event of the training sample. However the necessity to a priori define a positive scaling coefficient a still remains. In the current section we describe an enhancement of the learning algorithm that solves the latter, and allows to automatically adapt the scaling coefficient a to the training sample. It becomes possible by virtue of choosing the scaling coefficient from the condition that provides for the highest resolution of decision rule (1). In other words, – due to it's selection utilizing a condition of maximal width for the range of training sample events' membership degrees.

Define the minimal and maximal distances of the events from the learning sample to the "center" c – put

$$r^2 = \min_{n=1,...,N} (\varphi(x_n) - c)^2, \quad R^2 = \max_{n=1,...,N} (\varphi(x_n) - c)^2\,.$$

It's possible to show that for $0 < r < R$, the value

$$a = rR$$

is the unique positive value of the scaling coefficient that maximizes the range of the corresponding membership degrees of events from the training sample. The corresponding range is the segment

$$\left[\frac{r}{r + R}, \frac{R}{r + R}\right].$$

This maximal segment is centered with respect to $\frac{1}{2}$ and has a width of

$$\frac{R - r}{R + r}\,.$$

Thus in order to construct the adaptive (with respect to the training sample) scaling coefficient a – one should find the extreme point of the range for the training sample events' membership degrees. The corresponding coefficient is

$$a = a(w_1,..., w_N) = \sqrt{\min_{n=1,...,N}(\varphi(x_n) - c)^2 \cdot \max_{n=1,...,N}(\varphi(x_n) - c)^2}\,. \quad (3)$$

Putting this formula into the right part of the functional $J_1(c, w_1,..., w_N)$ definition, – we obtain the following functional:

$$J_2(c, w_1, ..., w_N) = \frac{1}{2} \cdot \sum_{i=1}^{N} \left(w_i^2 (\varphi(x_i) - c)^2 + a(w_1, ..., w_N)(1 - w_i)^2 \right) \cdot$$

Minimum point $(c, w_1, ..., w_N)$ of this functional, being inserted into expression (2), gives us a solution of the problem set in section II: the decision rule (1) is the corresponding membership function.

VII. Experiments, Applications and Commercial Implementations

In case of Gaussian kernel, the functional $J_1(c, w_1, ..., w_N)$ described in section V was considered in [6]. Verses this article, which is a natural extension of the least squares method, – approach taken in [6] is mainly based on ideas of SVM clusterization [8]. In the article [6] an experimental comparison of the SVM algorithm and the learning method, based on finding extreme of the functional $J_1(c, w_1, ..., w_N)$, was elaborated on. The LIBSVM [9] instruments and EPA-HTTP [10] data were utilized. Article [6] favors approach based on minimization of the functional $J_1(c, w_1, ..., w_N)$. However, the author has neither pointed to a specific value for the scaling coefficient *a* used nor provided an elaboration on any constructive reasons of its choice.

A number of applications using the learning algorithms based on minimization of the functional $J_1(c, w_1, ..., w_N)$, are known at the moment:

- evaluation of events in Data Mining [6],
- detection of certain network intrusions [11],
- spam protection [12].

But, as it was already mentioned above, the essential prior disadvantage for practical implementations of the learning algorithm, based on minimization of the functional $J_1(c, w_1, ..., w_N)$, was the necessity to a priori specify the scaling coefficient *a* for decision rule (1).

The adaptive learning algorithm suggested in this article is based on minimization of the functional $J_2(c, w_1, ..., w_N)$ and is free from that disadvantage. Indeed – now, after finding the minimum $(c, w_1, ..., w_N)$ of the functional $J_2(c, w_1, ..., w_N)$, the scaling coefficient *a* for decision rule (1) is computed automatically in accordance with expression (3). Moreover, since the choice of the scaling coefficient is based on the condition of the highest possible resolution of the rule (1), the suggested learning algorithm delivers the decision rule with the widest possible range of membership degrees.

On basis of the suggested adaptive algorithm, an effective concept of learning was designed for a wide range of input spaces. This concept was successfully implemented in several commercial software products, in particular:

- Adaptive Security Engine®: The formal commercial manifestation of the algorithms presented in this paper and others, and universal tool for data analysis [14].
- Adaptive Security Analyzer Pro®: Security Data Analysis application [15].
- Anomaly Analyzer®: Anomaly Analysis component of Quest Software data collection and reporting application, Intrust®, licensed from PWI, Inc./Privacyware [16].

VIII. Anomaly and Intrusion Detection Examples

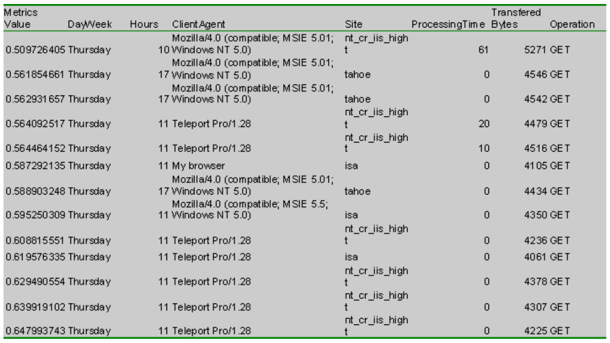
Two examples of applying the approach, based on minimization of the functional $J_2(c, w_1, ..., w_N)$, are given below.

Example 1. The approach was used to create a training baseline for events collected into a database by Quest Software InTrust® application. Typical events were represented by sets of monitored web-resources accessed from inside the organization (ISA WebProxy access).

In this case the input space Ω is represented by a set of multi-variable network events. Each events in a set has variables including DATE_TIME, SITE accessed (Web Resource), HOUR of the Day when SITE is accessed, Day of Week when Site is accessed, Transferred Bytes, Received bytes, Operation, and others.

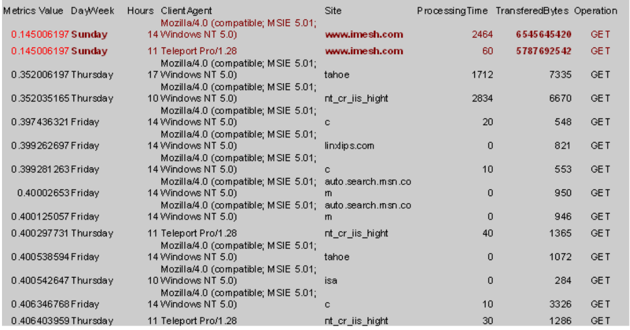
Detecting anomalies among Web resource requests is of interest to System/Security administrators responsible for the detection of security breaches, policy incompliance, misuse of System resources, vulnerability detection and other threats.

During the designated 'training period', Anomaly Analyzer (the commercial manifestation of the algorithmic approaches discussed in this paper) calculates a set of 'typical' events (a baseline). Some elements of the training sample include:



Metrics Value	DayWeek	Hours	ClientAgent	Site	ProcessingTime	Transfered Bytes	Operation
0.509726405	Thursday	10	Mozilla/4.0 (compatible; MSIE 5.01; Windows NT 5.0)	nt_cr_iis_high t	61	5271	GET
0.561854661	Thursday	17	Mozilla/4.0 (compatible; MSIE 5.01; Windows NT 5.0)	tahoe	0	4546	GET
0.562931657	Thursday	17	Mozilla/4.0 (compatible; MSIE 5.01; Windows NT 5.0)	tahoe	0	4542	GET
0.564092517	Thursday	11	Teleport Pro/1.28	nt_cr_iis_high t	20	4479	GET
0.564464152	Thursday	11	Teleport Pro/1.28	nt_cr_iis_high t	10	4516	GET
0.587292135	Thursday	11	My browser	isa	0	4105	GET
0.588903248	Thursday	17	Mozilla/4.0 (compatible; MSIE 5.01; Windows NT 5.0)	tahoe	0	4434	GET
0.595250309	Thursday	11	Mozilla/4.0 (compatible; MSIE 5.5; Windows NT 5.0)	isa	0	4350	GET
0.608815551	Thursday	11	Teleport Pro/1.28	nt_cr_iis_high t	0	4236	GET
0.619576335	Thursday	11	Teleport Pro/1.28	isa	0	4061	GET
0.629490554	Thursday	11	Teleport Pro/1.28	nt_cr_iis_high t	0	4378	GET
0.639919102	Thursday	11	Teleport Pro/1.28	nt_cr_iis_high t	0	4307	GET
0.647993743	Thursday	11	Teleport Pro/1.28	nt_cr_iis_high t	0	4225	GET

In practical use on large sets of actual data gathered from the networks, Anomaly Analyzer detected several events that represent 'unusual activity' for Web PROXY models. The table below indicates the event's deviation (metrics) value and the combination of variables that most influenced the event's classification. The most unusual events appear on top and are represented by a lower metrics value (0.145006197). In the sample below, Anomaly Analyzer indicates that the Day, resource accessed, and the volume of transferred bytes variables contributed most to the event's deviation from normal.

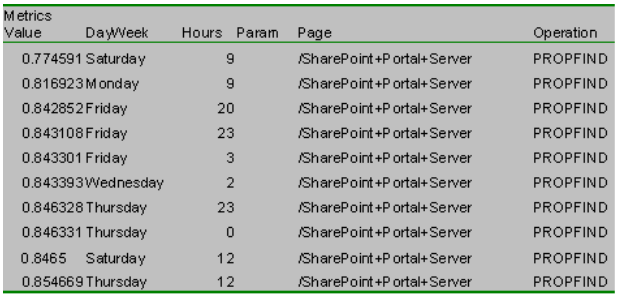


Metrics Value	DayWeek	Hours	ClientAgent	Site	ProcessingTime	TransferedBytes	Operation
0.145006197	**Sunday**	14	Mozilla/4.0 (compatible; MSIE 5.01; Windows NT 5.0)	**www.imesh.com**	2464	**6545645420**	GET
0.145006197	**Sunday**	11	Teleport Pro/1.28	**www.imesh.com**	60	**5787692542**	GET
0.352006197	Thursday	17	Mozilla/4.0 (compatible; MSIE 5.01; Windows NT 5.0)	tahoe	1712	7335	GET
0.352035165	Thursday	10	Mozilla/4.0 (compatible; MSIE 5.01; Windows NT 5.0)	nt_cr_iis_hight	2834	6670	GET
0.397436321	Friday	14	Mozilla/4.0 (compatible; MSIE 5.01; Windows NT 5.0)	c	20	548	GET
0.399262697	Friday	14	Mozilla/4.0 (compatible; MSIE 5.01; Windows NT 5.0)	linxlips.com	0	821	GET
0.399281263	Friday	14	Mozilla/4.0 (compatible; MSIE 5.01; Windows NT 5.0)	c	10	553	GET
0.40002653	Friday	14	Mozilla/4.0 (compatible; MSIE 5.01; Windows NT 5.0)	auto.search.msn.co m	0	950	GET
0.400125057	Friday	14	Mozilla/4.0 (compatible; MSIE 5.01; Windows NT 5.0)	auto.search.msn.co m	0	946	GET
0.400297731	Thursday	11	Teleport Pro/1.28	nt_cr_iis_hight	40	1365	GET
0.400538594	Friday	14	Mozilla/4.0 (compatible; MSIE 5.01; Windows NT 5.0)	tahoe	0	1072	GET
0.400542647	Thursday	10	Mozilla/4.0 (compatible; MSIE 5.01; Windows NT 5.0)	isa	0	284	GET
0.406346768	Friday	14	Mozilla/4.0 (compatible; MSIE 5.01; Windows NT 5.0)	c	10	3326	GET
0.406403959	Thursday	11	Teleport Pro/1.28	nt_cr_iis_hight	30	1286	GET

These events deviate measurably and significantly from the baseline established by Anomaly Analyzer. In this example, the deviations could represent a violation of system use policy. Anomaly Analyzer provides the security administrator an ability to detect this activity without the necessity of an explicit pre-configured rules-based mechanism. It provides security, compliance and system assurance practitioners with a more comprehensive capability to review large volumes of data and more effectively pinpoint and address potential system threats.

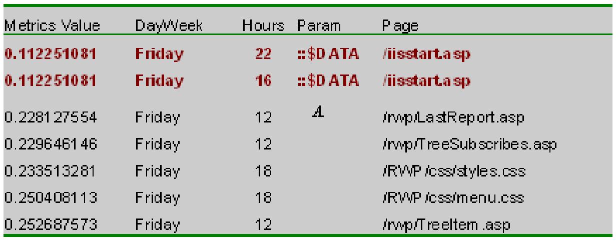
Example 2. In this case, anomaly analysis was performed using the Adaptive Security Analyzer on a database of Web Server (Microsoft IIS) events. Elements of 'input space Ω' have similar variables to Web Proxy example: DATE-TIME, Day of Week, Web PAGE (Site) requested, Processing Time, Transferred Bytes, Operation, and others.

For this example, the "Typical set of training sample events" included:



Metrics Value	DayWeek	Hours	Param	Page	Operation
0.774591	Saturday	9		/SharePoint+Portal+Server	PROPFIND
0.816923	Monday	9		/SharePoint+Portal+Server	PROPFIND
0.842852	Friday	20		/SharePoint+Portal+Server	PROPFIND
0.843108	Friday	23		/SharePoint+Portal+Server	PROPFIND
0.843301	Friday	3		/SharePoint+Portal+Server	PROPFIND
0.843393	Wednesday	2		/SharePoint+Portal+Server	PROPFIND
0.846328	Thursday	23		/SharePoint+Portal+Server	PROPFIND
0.846331	Thursday	0		/SharePoint+Portal+Server	PROPFIND
0.8465	Saturday	12		/SharePoint+Portal+Server	PROPFIND
0.854669	Thursday	12		/SharePoint+Portal+Server	PROPFIND

From the data in the table above, it is clear that access to SharePoint portal is 'typical' for this organization. Once other actual events were compared to the training sample, Adaptive Security Analyzer indicated that the following events most deviated from the normal baseline.



Metrics Value	DayWeek	Hours	Param	Page
0.112251081	**Friday**	**22**	**::$DATA**	**/iisstart.asp**
0.112251081	**Friday**	**16**	**::$DATA**	**/iisstart.asp**
0.228127554	Friday	12	A	/rwp/LastReport.asp
0.229646146	Friday	12		/rwp/TreeSubscribes.asp
0.233513281	Friday	18		/RWP /css/styles.css
0.250408113	Friday	18		/RWP /css/menu.css
0.252687573	Friday	12		/rwp/TreeItem .asp

The most unusual events (metrics value 0.112251081) represent a well known web server exploit. The algorithm enabled the detection and anomaly calculation to be performed on a dual-processor 3.2 GHZ server within a period less than 1.5 hour. The size of the database queried was over 50 Gigabytes.

IX. Conclusion

Adaptive, machine-learning approaches, such as those described herein can help improve and expand the value of anomaly intelligence applications (for enterprise security and other challenges).

The approach presented herein is an alternative to many common methods of support vector machines algorithms. It was conceived and implemented to enable the analysis of anomalous events to be performed adaptively, and with progressively improved accuracy and performance.

Acknowledgment

Authors express many thanks to Gregory J. Salvato for his help and useful comments.

References

[1] Herbrich R. Learning Kernel Classifiers: Theory and Algorithms. Cambridge, London, The MIT Press, 2002.

[2] Aizerman M.A., Braverman E.M., Rozonoer L.I. Theoretical foundations of the potential approach for margin classifiers. *Automation and Remote Control.* Vol. 25, P. 821-837, 1964.

[3] Schölkopf B., Smola A.J. Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond. Cambridge, London, The MIT Press, 2002.

[4] Haussler D. Convolutional kernels on discrete structures. *Technical Report UCSC-CRL-99-10*, Computer Science Department, University of California at Santa Cruz, 1999.

[5] Petrovskiy M.I. A Hybrid Method for Patterns Mining and Outliers Detection in the Web Usage Log. *AWIC'03*, P. 318-328, 2003.

[6] Petrovskiy M.I. Outlier Detection Algorithms in Data Mining Systems. *Programming and Computer Software*, Vol. 29, No. 4, P. 228-237. Springer Science – Business Media B.V., 2003.

[7] Ben-Hur A., Horn D., Siegelmann H.T., Vapnik V. Support Vector Clustering. *Journal of Machine Learning*, Vol. 2, P. 125-137, 2001.

[8] Chih-Chung Chang, Chih-Jen Lin. LIBSVM – a Library for Support Vector Machines, http://www.csie.ntu.edu.tw/~cjlin/libsvm

[9] Bottomley L. Dataset: a day of HTTP logs from the EPA WWW Server. Duke University, 1995, http://ita.ee.lbl.gov/html/contib/EPA-HTTP.html

[10] Petrovskiy M.I. A Fuzzy Kernel-Based Method for Real-Time Network Intrusion Detection . *IICS 2003*, P. 189-200.

[11] Mashechkin I., Petrovskiy M., Rozinkin A. Enterprise Anti-Spam Solution Based on Machine Learning Approach. *ICEIS (2) 2005*, P. 188-193.

[12] Adaptive Security Engine. Privacyware®, 2005, http://www.privacyware.com/ASE.html

[13] Adaptive Security Analyzer Pro. Privacyware®, 2005, http://www.privacyware.com/index_ASAPro.html

[14] InTrust® for Windows. Quest Software®, 2005, http://wm.quest.com/products/Intrust/