# EMBEDDED FEATURE SELECTION FOR SPAM AND PHISHING FILTERING USING SUPPORT VECTOR MACHINES

Sebastián Maldonado[1] and Gastón L'Huillier[2]
[1]*Faculty of Engineering and Applied Sciences, Universidad de los Andes*
*Av. San Carlos de Apoquindo 2200, Las Condes, Santiago, Chile*
[2]*Groupon, Inc., Palo Alto, CA, U.S.A.*

Keywords: Spam and phishing filtering, Support vector machines, Feature selection, Embedded methods.

Abstract: Today, the Internet is full of harmful and wasteful elements, such as phishing and spam messages, which must be properly classified before reaching end-users. This issue has attracted the pattern recognition community's attention and motivated to determine which strategies achieve best classification results. Several methods use as many features as content-based properties the data set have, which leads to a high dimensional classification problem. In this context, this paper presents a feature selection approach that simultaneously determines a non-linear classification function with minimal error and minimizes the number of features by penalizing their use in the dual formulation of binary Support Vector Machines (SVM). The method optimizes the width of an anisotropic RBF Kernel via successive gradient descent steps, eliminating features that have low relevance for the model. Experiments with two real-world Spam and Phishing data sets demonstrate that our approach accomplishes the best performance compared to well-known feature selection methods using consistently a small number of features.

## 1 INTRODUCTION

One particular domain for which machine learning has been considered a key component is cyber-security. Specifically, for the correct identification of the large number of spam messages, web spam, and spam servers which inundate Internet resources every day. It is likely that spam messages will continue to be one of the most wasteful, dangerous and infectious elements on the Web as new campaigns are occasionally instigated by spam senders (Taylor et al., 2007).

Identifying malicious emails such as spam or phishing can be considered as a task of binary classification where the goal is to discriminate between the two classes of "desired" and "undesired" emails. Support Vector Machine (Vapnik, 1998) is an effective classification method and provides several advantages such as absence of local minima, adequate generalization to new objects, and representation that depends on few parameters. Furthermore, this method has proved to be very effective for spam classification (Tang et al., 2008) and Phishing (L'Huillier et al., 2010). However, this approach does not determine the importance of the features used by a classifier (Maldonado and Weber, 2009). In this paper we present a feature selection approach for binary classification using SVM, showing its potential for spam and phishing classification.

This paper is organized as follows. In Section 2 we briefly introduce spam and phishing classification. Recent developments for feature selection using SVM are reviewed in Section 3. Section 4 presents the proposed feature selection method based on SVM. Experimental results using real-world data sets are given in Section 5. A summary of this paper can be found in Section 6, where we provide its main conclusions and address future developments.

## 2 SPAM AND PHISHING CLASSIFICATION

Among all counter-measures used against spam and phishing, there are two main alternatives (Bergholz et al., 2010): content-based classification on the one hand and blacklisting and white-listing on the other. In the following, the main approaches for these alternatives are briefly reviewed.



Maldonado S. and L'Huillier G. (2012).
EMBEDDED FEATURE SELECTION FOR SPAM AND PHISHING FILTERING USING SUPPORT VECTOR MACHINES.
In *Proceedings of the 1st International Conference on Pattern Recognition Applications and Methods*, pages 445-450
DOI: 10.5220/0003782004450450

### 2.1 Content-based Classification

Spam filtering is a classical problem in machine learning, and many filtering techniques have been described (Goodman et al., 2007). However, in terms of content-based classification, phishing differs in many aspects from the spam case. While most of spam emails are intended to spread information about products and web sites, in phishing, the interaction between a message and the receiver is more complex. End users are usually involved in a third step of interaction, such as following malicious links, filling deceptive forms, or replying with useful information which are relevant for the fraud message to succeed.

Previous works on content-based filtering of deceptive spam or phishing emails have focused on the extraction of a large number of features used in popular machine learning techniques for its classification (Bergholz et al., 2010).

### 2.2 Network-based Classification

Real Time Blacklists (RBLs) have been considered as an efficient alternative to filtering spam messages, just by considering server-side features for spam sender detection. These services can be queried over the Domain Name System (DNS) protocol, which provides a powerful tool for email servers to decide whether or not to accept messages from a given host (Tang et al., 2008). These approaches are based on features extracted from network properties and not from content-based characteristics, hence the dimensionality of the classification problem is considerably low and the features' properties are different than in content-based approaches. For this reason, these approaches were not considered in this paper.

## 3 EMBEDDED FEATURE SELECTION FOR SVMS

There are different strategies for embedded feature selection. First, feature selection can be seen as an optimization problem. For example, the methods presented in (Neumann et al., 2005) add an extra term that penalizes the cardinality of the selected feature subset to the standard cost function of SVM. By optimizing this modified cost function features are selected simultaneously to model construction. Another embedded approach is the Feature Selection ConcaVe (FSV) (Bradley and Mangasarian, 1998), based on the minimization of the "zero norm" : $\|\mathbf{w}\|_0 = |\{i : w_i \neq 0\}|$. Note that $\|\cdot\|_0$ is not a norm because the triangle inequality does not hold (Bradley and Mangasarian, 1998), unlike $l_p$-norms with $p > 0$. Since the $l_0$-"norm" is non-smooth, it was approximated by a concave function:

$$\|\mathbf{w}\|_0 \approx \mathbf{e}^T(\mathbf{e} - \exp(-\beta|\mathbf{w}|) \quad (1)$$

with an approximation parameter $\beta \in \mathbb{R}_+$ and $\mathbf{e} = (\mathbf{1},\ldots,\mathbf{1})^{\mathbf{T}}$. The problem is finally solved by using an iterative method called Successive Linearization Algorithm (SLA) for FSV (Bradley and Mangasarian, 1998). (Weston et al., 2003) proposed an alternative approach for zero-"norm" minimization ($l_0$-SVM) by iteratively scaling the variables, multiplying them by the absolute value of the weight vector $\mathbf{w}$. An important drawback of these methods is that they are limited to linear classification functions (Guyon et al., 2006).

Several embedded approaches consider backward feature elimination in order to establish a ranking of features, using SVM-based contribution measures to evaluate their relevance. One popular method is known as Recursive Feature Elimination (SVM-RFE) (Guyon et al., 2009). The goal of this approach is to find a subset of size $r$ among $n$ variables ($r < n$) which maximizes the classifier's performance. The feature to be removed in each iteration is the one whose removal minimizes the variation of $W^2(\alpha)$:

$$W^2(\alpha) = \sum_{i,s=1}^{m} \alpha_i \alpha_s y_i y_s K(\mathbf{x}_i, \mathbf{x}_s) \quad (2)$$

The scalar $W^2(\alpha)$ is a measure of the model's predictive ability and is inversely proportional to the margin. Features are eliminated applying the following procedure:

1. Given a solution $\alpha$, for each feature $p$ calculate:

$$W^2_{(-p)}(\alpha) = \sum_{i,s=1}^{m} \alpha_i \alpha_s y_i y_s K(\mathbf{x}_i^{(-p)}, \mathbf{x}_s^{(-p)}) \quad (3)$$

   where $\mathbf{x}_i^{(-p)}$ represents the training object $i$ with feature $p$ removed.

2. Eliminate the feature with smallest value of $|W^2(\alpha) - W^2_{(-p)}(\alpha)|$.

Another ranking method that allows kernel functions was proposed in (Rakotomamonjy, 2003), which considers a *leave-one-out* error bound for SVM, the *radius margin bound* (Vapnik, 1998) $LOO \leq 4R^2||\mathbf{w}||^2$, where $R$ denotes the radius of the smallest sphere that contains the training data. This bound is also used in (Weston et al., 2001) through the *scaling factors* strategy. Feature selection is performed by scaling the input parameters by a vector $\sigma \in [0,1]^n$.

Large values of $\sigma_j$ indicate more useful features. The problem consists in choosing the best kernel of the form:

$$K_\sigma(\mathbf{x}_i, \mathbf{x}_s) \equiv K(\sigma * \mathbf{x}_i, \sigma * \mathbf{x}_s) \quad (4)$$

where $*$ is the component-wise multiplication operator. The method presented by (Weston et al., 2001) considers the gradient descent algorithm for updating $\sigma$. (Canu and Grandvalet, 2002) propose to limit the use of the attributes by constraining the scaling factors using a parameter $\sigma_0$, which controls the norm of $\sigma$.

# 4 THE PROPOSED METHOD FOR EMBEDDED FEATURE SELECTION

An embedded method for feature selection using SVMs is proposed in this section. The main idea is to penalize the use of features in the dual formulation of SVMs using a gradient descent approximation for Kernel optimization and feature elimination. The proposed method attempts to find the best suitable RBF-type Kernel function for each problem with a minimal dimension by combining the parameters of generalization (using the 2-norm), goodness of fit, and feature selection (using a 0-"norm" approximation).

For this approach we use the anisotropic Gaussian Kernel:

$$K(\mathbf{x}_i, \mathbf{x}_s, \sigma) = exp\left(-\frac{||\sigma * \mathbf{x}_i - \sigma * \mathbf{x}_s||^2}{2}\right) \quad (5)$$

where $*$ denotes the component-wise vector product operator, defined as $\mathbf{a}*\mathbf{b} = (a_1 b_1, \ldots, a_n b_n)$.

The proposed approach (Kernel-Penalized SVM) incorporates feature selection in the dual formulation of SVMs. The formulation includes a penalization function $f(\sigma)$ based on the 0-"norm" approximation (1) described in Section 3 and modifying the Gaussian Kernel using an (anisotropic) width vector $\sigma$ as a decision variable. The feature penalization should be negative since the dual SVM is a maximization problem. The following embedded formulation of SVMs for feature selection is proposed:

$$\underset{\alpha,\sigma}{\text{Max}} \quad \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,s=1}^{m} \alpha_i \alpha_s y_i y_s K(\mathbf{x}_i, \mathbf{x}_s, \sigma) - C_2 f(\sigma) \quad (6)$$

subject to

$$\sum_{i=1}^{m} \alpha_i y_i = 0$$

$$0 \leq \alpha_i \leq C \qquad i \in \{1, \ldots, m\}.$$

$$\sigma_j \geq 0 \qquad j \in \{1, \ldots, n\}.$$

Notice that the values of $\sigma$ are always considered to be positive, in contrast to the weight vector $\mathbf{w}$ in formulation (1), since it is desirable that the kernel widths be positive values (Maldonado et al., 2011). Considering the "zero norm" approximation described in (1), $\|\sigma\|_0 \approx \mathbf{e}^T(\mathbf{e} - \exp(-\beta|\sigma|)$, and since $|\sigma_j| = \sigma_j \ \forall j$, it is not necessary to use the 1-norm in the approximation.

The following feature penalization function is proposed, where the approximation parameter $\beta$ is also considered. In (Bradley and Mangasarian, 1998), the authors suggest setting $\beta$ to 5:

$$f(\sigma) = \mathbf{e}^T(\mathbf{e} - \exp(-\beta\sigma) = \sum_{j=1}^{n} [1 - exp(-\beta\sigma_j)] \quad (7)$$

Since the formulation (6) is non-convex, we develop an iterative algorithm for its approximation. A 2-step methodology is proposed: first we solve the traditional dual formulation of SVM for a fixed anisotropic kernel width $\sigma$:

$$\underset{\alpha}{\text{Max}} \quad \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,s=1}^{m} \alpha_i \alpha_s y_i y_s K(\mathbf{x}_i, \mathbf{x}_s, \sigma) \quad (8)$$

subject to

$$\sum_{i=1}^{m} \alpha_i y_i = 0$$

$$0 \leq \alpha_i \leq C \qquad i \in \{1, \ldots, m\}.$$

In the second step the algorithm solves, for a given solution $\alpha$, the following non-linear formulation:

$$\underset{\sigma}{\text{Min}} \quad F(\sigma) = \sum_{i,s=1}^{m} \alpha_i \alpha_s y_i y_s K(\mathbf{x}_i, \mathbf{x}_s, \sigma) + C_2 f(\sigma) \quad (9)$$

subject to

$$\sigma_j \geq 0 \qquad j \in \{1, \ldots, n\}.$$

The goal of formulation (9) is to find a sparse solution, making zero as many components of $\sigma$ as possible. We propose an iterative algorithm that updates the anisotropic kernel variable $\sigma$, using the gradient of the objective function, and eliminates the features that are close to zero (below a given threshold $\varepsilon$). The algorithm solves successive gradient descent steps until one particular scaling factor $\sigma_j$ drops below a threshold $\varepsilon$, starting with one initial solution $\sigma_0$. When this happens, attribute $j$ is eliminated by setting $\sigma_j = 0$.

**Algorithm 1:** Kernel Width Updating and Feature Elimination.

1. Start with $\sigma = \sigma_0$;
2. flag=true; flag2=true;
3. **while**(flag==true) **do**
4. train SVM (formulation (8));
5. $t = 0$;
6. **while**(flag2==true) **do**
7. $\sigma^{t+1} = \sigma^t - \gamma \Delta F(\sigma^t)$;
8. **if** $(||\sigma^{t+1} - \sigma^t||_1 < \varepsilon')$ **then**
9. flag2==false, flag==false;
10. **else**
11. **if** $(\exists j \mid \sigma_j^{t+1} > 0 \wedge \sigma_j^{t+1} < \varepsilon, \forall j)$ **then**
12. **for all** $(\sigma_j^{t+1} < \varepsilon)$ **do** $\sigma_j^{t+1} = 0$;
13. flag2==false;
14. **end if**
15. **end if**
16. $t = t + 1$;
17. **end while**;
18. **end while**;

Then the algorithm returns to formulation (8) until convergence. It is also possible that several variables become zero in one iteration. The algorithm Kernel Width Updating and Feature Elimination follows:

In the seventh line the algorithm adjusts the Kernel variables by using the gradient descent procedure, incorporating a gradient parameter $\gamma$. In this step the algorithm computes the gradient of the objective function in formulation (9) for a given solution of SVMs $\alpha$, obtained by training an SVM classifier using formulation (8). For a given feature $j$, the gradient of formulation (9) is:

$$\Delta_j F(\nu) = C_2 \beta exp(-\beta \sigma_j) \qquad (10)$$
$$+ \sum_{i,s=1}^{m} \sigma_j (x_{i,j} - x_{s,j})^2 \alpha_i \alpha_s y_i y_s K(\mathbf{x}_i, \mathbf{x}_s, \sigma)$$

Lines 8 and 9 of the algorithm represent the stopping criterion, which is reached when $\sigma^{t+1} \approx \sigma^t$. Lines 11 to 14 of the algorithm represent the feature elimination step. When a Kernel variable $\sigma_j$ in iteration $t+1$ is below a threshold $\varepsilon$, this feature is considered as irrelevant and eliminated by setting $\sigma_j = 0$. This variable will not be included in subsequent iterations of the algorithm.

# 5 RESULTS FOR SPAM AND PHISHING DATA SETS

We applied the proposed approach for feature selection to two data sets. We consider the following procedure for model comparison: First, model selection is performed before feature selection, obtaining the kernel parameters $d$, $\rho$ and penalty parameter $C$. The best combination is selected via 10-fold cross-validation. For the methods RFE-SVM, FSV-SVM and Fisher Filtering a ranking is first obtained with the training data, and model performance is then obtained using 10-fold cross-validation for specific numbers of attributes, depending on the size of the data set, considering the hyper-parameters obtained during the model selection procedure. For KP-SVM, instead, the algorithm runs using initial hyper-parameters and automatically obtains the desired number of features and the Kernel shape when convergence is reached we compute also the average cross-validation performance in intermediate steps for comparison purposes. The parameters for KP-SVM were selected previously according to the following values:

- Parameter $C_2$ represents the penalty for the feature usage and is strongly related to $C$, the original regularization parameter. $C_2$ is considered the most important parameter for KP-SVM, since classification results change significantly varying its values. We try the following values, monitoring both classification accuracy and feature usage: $C_2 = \{0, 0.5C, C, 2C\}$
- The initial (isotropic) kernel width $\sigma_0$, the threshold $\varepsilon$ and the gradient parameter $\gamma$ are considered less influential in the final solution, according to our empirical results. We set $\sigma_0 = \frac{1}{\rho^2} \cdot \mathbf{e}$, where $\rho$ is the isotropic kernel width obtained in a previous step for model selection considering all features, and $\mathbf{e}$ is a vector of ones of the size of the number of current features in the solution; $\varepsilon = \frac{1}{100\rho^2}$ and $\gamma = 0.1\varepsilon||\Delta F(\sigma^0)||$, where $||\Delta F(\sigma^0)||$ represents the Euclidean norm of the first computed gradient vector. This combination of parameters guarantees both a sufficiently small $\varepsilon$ that avoids the removal of relevant features and an adequate update of the kernel variables, controlled by the magnitude of the components of $\Delta F(\sigma)$. This parameter avoids a strong fluctuation of the kernel variables and negative widths, especially at the first iterations of the algorithm.

## 5.1 Description of Data Sets

In this subsection we briefly describe the different data sets mentioned above.

**Spambase Data Set (Spam).** The Spambase Data set from the UCI data repository (Asuncion and Newman, 2007) presents 57 features and 4,601 instances (2,788 emails labeled as spam and 1,813 ham[1] emails). The data set was created by Mark Hopkins, Erik Reeber, George Forman and Jaap Suermondt from the Hewlett Packard Labs.

Most of the features indicate whether a particular word or character was frequently occurring in the email. The data set presents 48 continuous attributes representing the percentage of words in the email that match a particular word, 6 continuous attributes representing the percentage of characters in the email that match a particular character, the average length of uninterrupted sequences of capital letters, the length of the longest uninterrupted sequences of capital letters and the total number of capital letters in the email. The predictive variables were scaled between 0 and 1.

**Phishing Data Set (Phishing).** The phishing corpus used to test the proposed methodology, was an English language phishing email corpus built using Jose Nazario's phishing corpus[2] and the SPAMASSASSIN ham collection. The phishing corpus consists of 4,450 emails manually retrieved from November 27, 2004 to August 7, 2007.

The ham corpus was built using the Spamassassin collection, from the Apache SPAMASSASSIN Project[3], based on a collection of 6,951 ham email messages. Both phishing and ham messages are available in UNIX mbox format. All features were extracted according to (L'Huillier et al., 2010).

## 5.2 Results using Kernel-penalized Feature Selection

First we compare the results of the best model found using the described model selection procedure for the three different kernel functions: linear, polynomial, and Gaussian kernel. The following set of values for the parameters (penalty parameter $C$, degree of the polynomial function $d$ and Gaussian Kernel width $\sigma$) were used (Maldonado and Weber, 2009):
$C = \{0.1, 0.5, 1, 10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 200, 300, 400, 500, 1000\}$
$d = \{2, 3, 4, 5, 6, 7, 8, 9\}$
$\rho = \{0.1, 0.5, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 20, 100\}$.

Best cross-validation results were achieved for both data sets using the Gaussian Kernel. Then we compared the classification performance of the different ranking criteria for feature selection by plotting the mean test accuracy for an increasing number of ranked features used for learning. Figures 1 and 2 show the results for each data set respectively. The proposed KP-SVM approach provides only the information until the stopping criterion is reached.

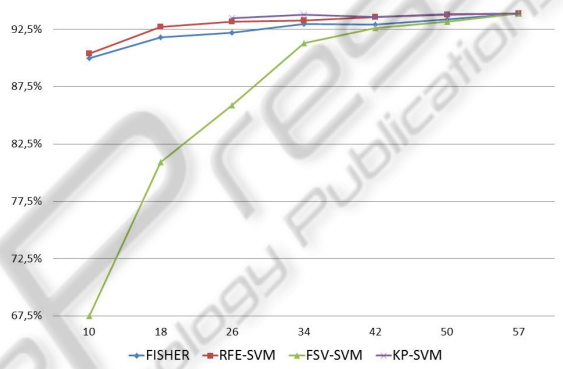


Figure 1: Mean of test accuracy for Spam vs. the number of ranked variables used for training.

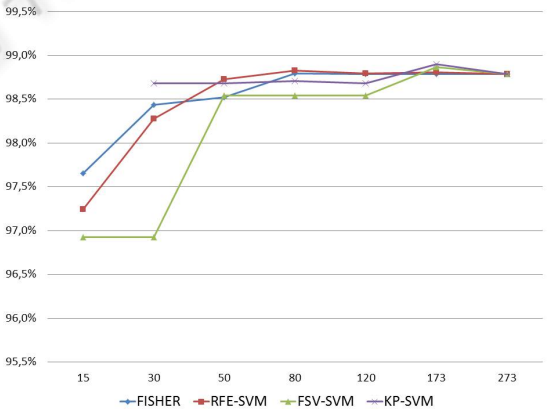


Figure 2: Mean of test accuracy for Phishing vs. the number of ranked variables used for training.

These experiments underline that the proposed approach, KP-SVM, outperforms other feature selection methods in terms of classification performance for a small number of features in both data sets used. Another important remark is that best classification performance is achieved for KP-SVM considering $C_2 = C$ for the Spam data set and $C_2 = 0.5C$ for the

[1]"Ham" is the name used to describe regular messages that are neither spam nor phishing.

[2]Available at http://bit.ly/jnazariophishing [Online: accesses November 02, 2011].

[3]Available at http://spamassassin.apache.org/publiccorpus/ [Online: accessed November 02, 2011].

Phishing data set. For both data sets the use of feature penalization outperforms the model obtained using $C_2 = 0$, which can be considered a variant of the ARD model presented in (Chapelle et al., 2002). This fact proves the importance of feature selection in relatively high dimensional data sets, such as the ones presented in this work.

## 6 CONCLUSIONS

In this work we present an embedded approach for feature selection using SVM applied to phishing and spam classification. A comparison with other feature selection methods shows its advantages:

- It outperforms other techniques in terms of classification accuracy.
- It is not necessary to set *a priori* the number of features to be selected, unlike other feature selection approaches. The algorithm determines the optimal feature number according to the regularization parameter $C_2$.
- It can be used with other kernel functions, such as linear and polynomial kernels.

Even if several parameters have to be tuned, the computational effort can be reduced since the search for a feature subset can be obtained automatically, reducing computational time by avoiding a validation step on finding an adequate number of features.

Future work has to be done in various directions. First, we consider the extension to highly imbalanced data sets, a very relevant topic in phishing and spam classification, and in pattern recognition in general. Furthermore, the current scenario for spam and phishing classification suggests the extension of the proposed embedded feature selection technique to very large databases as an important research opportunity.

## ACKNOWLEDGEMENTS

Support from the Chilean "Instituto Sistemas Complejos de Ingeniería" (ICM: P-05-004-F) is greatly acknowledged.

## REFERENCES

Asuncion, A. and Newman, D. (2007). UCI machine learning repository.

Bergholz, A., Beer, J. D., Glahn, S., Moens, M.-F., Paass, G., and Strobel, S. (2010). New filtering approaches for phishing email. *Journal of Computer Security*, 18(1):7–35.

Bradley, P. and Mangasarian, O. (1998). Feature selection vía concave minimization and support vector machines. In *Int. Conference on Machine Learning*, pages 82–90.

Canu, S. and Grandvalet, Y. (2002). Adaptive scaling for feature selection in SVMs. *Advances in Neural Information Processing Systems*, 15:553–560.

Chapelle, O., Vapnik, V., Bousquet, O., and Mukherjee, S. (2002). Choosing multiple parameters for support vector machines. *Machine Learning*, 46:131–159.

Goodman, J., Cormack, G. V., and Heckerman, D. (2007). Spam and the ongoing battle for the inbox. *Commun. ACM*, 50(2):24–33.

Guyon, I., Gunn, S., Nikravesh, M., and Zadeh, L. A. (2006). *Feature extraction, foundations and applications*. Springer, Berlin.

Guyon, I., Saffari, A., Dror, G., and Cawley, G. (2009). Model selection: Beyond the bayesian frequentist divide. *Journal of Machine Learning research*, 11:61–87.

L'Huillier, G., Hevia, A., Weber, R., and Rios, S. (2010). Latent semantic analysis and keyword extraction for phishing classification. In *ISI'10: Proceedings of the IEEE International Conference on Intelligence and Security Informatics*, pages 129–131, Vancouver, BC, Canada. IEEE.

Maldonado, S. and Weber, R. (2009). A wrapper method for feature selection using support vector machines. *Information Sciences*, 179:2208–2217.

Maldonado, S., Weber, R., and Basak, J. (2011). Kernel-penalized SVM for feature selection. *Information Sciences*, 181(1):115–128.

Neumann, J., Schnörr, C., and Steidl, G. (2005). Combined svm-based feature selection and classification. *Machine Learning*, 61:129–150.

Rakotomamonjy, A. (2003). Variable selection using SVM-based criteria. *Journal of Machine Learning research*, 3:1357–1370.

Tang, Y., Krasser, S., Alperovitch, D., and Judge, P. (2008). Spam sender detection with classification modeling on highly imbalanced mail server behavior data. In *Proceedings of the International Conference on Artificial Intelligence and Pattern Recognition*, AIPR'08, pages 174–180. ISRST.

Taylor, B., Fingal, D., and Aberdeen, D. (2007). The war against spam: A report from the front line. In *In NIPS 2007 Workshop on Machine Learning in Adversarial Environments for Computer Security*.

Vapnik, V. (1998). *Statistical Learning Theory*. John Wiley and Sons.

Weston, J., Elisseeff, A., Schölkopf, B., and Tipping, M. (2003). The use of zero-norm with linear models and kernel methods. *Journal of Machine Learning research*, 3:1439–1461.

Weston, J., Mukherjee, S., Chapelle, O., Ponntil, M., Poggio, T., and Vapnik, V. (2001). Feature selection for SVMs. In *Advances in Neural Information Processing Systems 13*, volume 13.