Association for Information Systems
AIS Electronic Library (AISeL)

ICIS 2010 Proceedings

International Conference on Information Systems (ICIS)

2010

A SEQUENTIAL MODEL FOR GLOBAL SPAM-CLASSIFYING PROCESSES

Wolfgang R. Burkart
University of Augsburg, wolfgang.burkart@wiwi.uni-augsburg.de

Stefan Etschberger
University of Applied Sciences, ste@hs-weingarten.de

Christian Klein
University of Hohenheim, cklein@uni-hohenheim.de

Dennis Kundisch
University of Paderborn, dennis.kundisch@wiwi.uni-paderborn.de

A SEQUENTIAL MODEL FOR GLOBAL SPAM-CLASSIFYING PROCESSES

Research-in-Progress

Wolfgang R. Burkart
Chair of Analytics & Optimization
University of Augsburg
86159 Augsburg, Germany
wolfgang.burkart@wiwi.uni-augsburg.de

Stefan Etschberger
Hochschule Ravensburg-Weingarten
University of Applied Sciences
88250 Weingarten, Germany
ste@hs-weingarten.de

Christian Klein
Chair of Accounting and Finance (510c)
University of Hohenheim
70593 Stuttgart, Germany
cklein@uni-hohenheim.de

Dennis Kundisch
Chair of Information Management
& E-Finance
University of Paderborn
33095 Paderborn, Germany
dennis.kundisch@wiwi.upb.de

Abstract

No current single filtering algorithm used to identify spam can provide for an error rate of zero. Different filtering approaches vary in technical and algorithmic aspects resulting in different error rates and costs to accomplish the classification goal. Therefore it is common practice in larger organizations to implement a spam-classifying process consisting of different single filters. We suggest a general model that aggregates cost and profit parameters of each filter step to an output, which represents the goodness of the whole classifying process. Optimizing this non-linear function leads to a problem which can be addressed by a heuristic approach.

Keywords: Algorithms, Spam, Email, Decision Models, Information Management, IT Security

Introduction

A problem nearly every internet user is confronted with is unsolicited (commercial) emails: emails the recipient does not want – better known as *spam*. The deletion of a single spam-email takes merely a couple of seconds for the user, therefore spam was considered as a personal annoyance for years. In the meantime this phenomenon causes costs amounting to more than $50 billion (Van Alstyne 2007). The impact becomes clear when one considers the ratio of spam to total emails in the internet, which has according to many sources grown from circa 30% in the beginning of 2003 (Lueg 2003) to mostly above 80% since 2005 (Lueg et al. 2007; MessageLabs 2010).

Due to the absence of a precise technical definition of spam, developing anti-spam measures is technically challenging. The problem is that by definition the nature of spam depends on the recipient's attitude towards receiving respective messages (Lueg et al. 2007; Lueg 2005). This is the major reason why a lot of varying approaches have been used and proposed against spam recently. These approaches include *origin-based filters* using whitelists, blacklists or realtime-blackhole-lists (Bager 2003) as well as greylists (González-Talaván 2006) and *content-based filters* using pattern matching (Lueg 2003), Bayesian probability (Androutsopoulos et al. 2000; Graham 2004), case-based-techniques (Delany et al. 2004) or a neural-network (Cao et al. 2004). These filtering methods differ in a lot of aspects, which can be summarized as costs per filtered email, rate of falsely classified spam (*false negatives*) and rate of falsely classified solicited emails (*false positives*). Spam handling in a corporate environment therefore requires adopting an optimal cost-benefit filter strategy including decisions about number, types and intensity of the filter mechanisms. While there are a lot of contributions that suggest specific filtering strategies (e.g. Androutsopoulos et al. 2000; Bager 2003; Cao et al. 2003; Carpinter and Hunt 2006; Delany et al. 2004; Fdez-Riverola et al. 2007; González-Talaván 2006; Graham 2004; Macmillan and Creelman 2005 and references therein), a lack of recommendations for the CIO how to decide about the application of these methods – stand-alone or in combination – can be identified in the literature.

This is somewhat surprising since it has been suggested in several contributions that combinations of different methods are needed. For example (Cranor and LaMacchia 1998) broadly concluded with respect to the rising spam problem more than 10 years ago: "[…] we recommend that user-friendly email software be developed that supports a multi-pronged technical solution […]." Likewise (Leiba and Borenstein 2004) emphasize: "While a single classification mechanism can provide a reasonable barrier to spam, any one algorithm is more easily defeated. Instead, an amalgamation of techniques may be used, giving multiple levels of classification and providing a better, more attack-proof shield to spam." And recently in the adjacent research area of Internet abuse, Chou et al. find that the performance of Internet filters and web page classifiers built using six text-mining algorithms differ substantially. Underpinning the demand for integrated solutions, the authors state: "Another interesting future direction would be to design methods for combining multiple (source-based and content-based) techniques to further improve performance." (Chou et al. 2010).

In this contribution, a model for a global spam-classifying process is suggested that is capable of integrating an arbitrary number of filter levels and associated filtering strategies. We focus on direct effects of spam on organizations. Secondary effects of anti-spam measures, like disproportionately affecting the legitimate emails of certain disempowered groups (also denoted as "digital redlining"), although important as discussed in (Lueg et al. 2007), will not be the issue here. Moreover, we focus our interest on the combination of technical means an organization can undertake to deal with spam. Thereby we abstract from regulatory means (as discussed e.g. in (Cranor and LaMacchia 1998)) as well as from suggested economic means (Van Alstyne 2007; Loder et al. 2006) to deal with spam.

We apply a combination of an evolutionary strategy with the quasi-Newton method known as the Broyden-Fletcher-Goldfarb-Shanno (BFGS) method to improve local best solutions faster to a nearby local optimum. Recommendations for the optimal intensity of each filter within a given architecture of sequential filters can be derived with our model. The applicability of the proposed model is visualized using an example with three sequential filters. The paper is organized as follows: In section 2 the model is presented. Section 3 proposes a solution for the model, which is clarified with an example in section 4. Finally, the main findings as well as prospects for further research are summarized in section 5.

Model

The following terms are used in this contribution: *real positives* for non-spam emails that are correctly classified, *real negatives* for spam email that are correctly classified, *false positives* for non-spam emails incorrectly classified as spam and *false negatives* for spam email incorrectly classified as non-spam email. Note that in signal-detection theory, spam emails would be classified as true positives or real positives (Green et al. 1966; Macmillan and Creelman 2005), since spam emails belong to the targeted population. We follow the notation of the specific spam literature (Cao et al. 2004; Graham 2004; Lueg 2003) here that considers spam (real negatives) to be associated with something negative, which is consistent with the general intuition when talking about spam.

As all common mail filter types are intended to work in a certain order, a sequential model seems to be an appropriate form of describing the flow of emails passing an institution's or a company's intranet. Figure 1 illustrates the emails' way coming from the internet until reaching the individual recipient. Each node in this sequence is characterized by its individual filter type in combination with a parameter describing the amount or the intensity within which this specific filter is used. This parameter ranges from 0% to 100% and represents the impact the filter has on the passing email resulting in success rates and failure rates, respectively. In consequence, each filter node results in certain costs caused by classifying the incoming emails, which can be for example computational costs, costs resulting from administrative effort put into this type of filter or licensing expenditures.

For instance, the intensity for a blacklist filter can be varied by the ordered subscription. To name just the extremes, there are free blacklists available on the one hand and hourly updated blacklists subject to a substantial subscription charge on the other hand. Moreover, the number of blacklist databases that are searched when applying a specific blacklist filter may be used to determine the level of intensity. The filter itself works in the same way in both cases; however, the success rates as well as the associated costs will differ.

Analogously, a Bayesian filter might be configured to e.g. either just analyze the subject line, the first 50 characters, the first 100 characters or the whole message body. This will also translate into different levels of intensity (e.g. 10%, 30%, 50% and 100%) and different associated costs (e.g. in terms of consumed CPU time for the analysis).

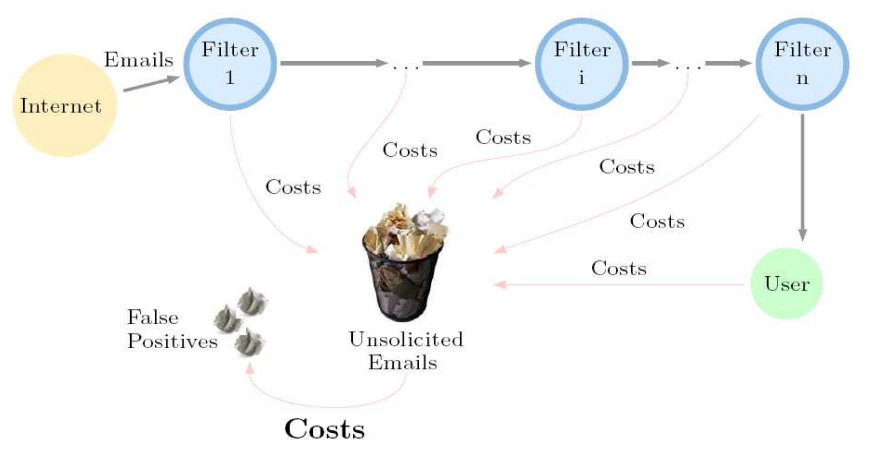

Figure 1. Schema of a filtering process

The success and cost rates for each filter i subject to its filter intensity $a_i \in [0,1]$ lead to a rate of falsely classified spam f_i^N, a rate of falsely classified solicited emails f_i^P, absolute costs per filter C_i^{fix} and relative costs per filtered email c_i^{rel} (with C_i as total costs of filter i). Apparently the configuration of such a multi-level spam filtering system requires the help of an experienced system administrator, since it is not only the separate filters that have to be calibrated but the system architecture as a whole. Even different positions of the same two filters in the sequence can and often will change the input and output structure of the average spam distribution of each component. This should be considered when estimating the rates f_i^N and f_i^P for the calibration of the overall system. These interdependencies of the filter rates subject to the position with the overall filtering system make the optimization task quite complex and are beyond the scope of this contribution. For the sake of simplification and for the formal treatment of the problem, we assume in the following that the filter rates can be determined independently from the position in the overall system.

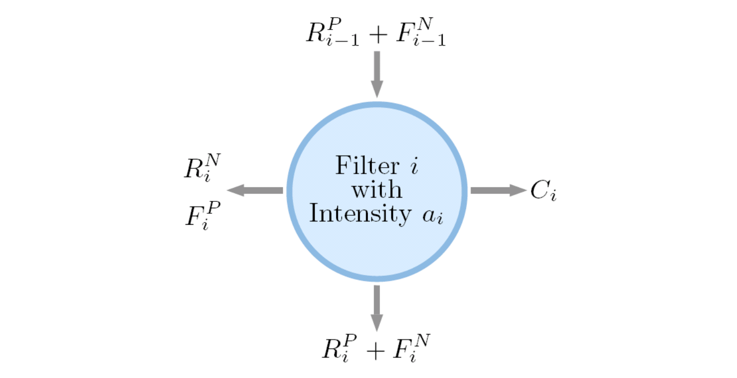

Figure 2. Schema of one specific filter node

In this model, the recipient of the remaining emails that passed through the preceding filters is regarded as the last filter in the sequence having its own cost and classification rates. Figure 2 shows an illustration of one filter node with respect to the number of filtered emails and the resulting total costs C_i of filter *i*.

The general model of one filter node is described as follows: All emails at the node input have been classified as wanted by the preceding nodes. That includes the absolute number of real positives R^P_{i-1} together with mails that have been falsely rated as wanted counted by F^N_{i-1}. Inside filter *i* the incoming emails are filtered depending on the intensity parameter a_i. This is modeled by filter-specific intensity-dependent functions resulting in the rate of the falsely classified unwanted emails as with $f_i^N : [0,1] \to [0,1]$ and the rate of the falsely classified wanted emails through $f_i^P : [0,1] \to [0,1]$. The relative filter rates f_i^N are limited to monotonic decreasing functions with $f_i^N(0) = 1$, which represent the behavior of a filter completely switched off and therefore "classifying" all incoming unsolicited emails as wanted. The rate functions of the falsely classified wanted emails f_i^P are restricted to monotonic increasing functions with $f_i^P(0) = 0$ and $f_i^P(1) = f^P_{i\,\max}$ with a considerable low maximum rate $f^P_{i\,\max}$. The number of emails classified as wanted by node *i* represented by the output of node *i* going to the next filter summarizes therefore to an absolute number

$$R_i^P + F_i^N = \left(1 - f_i^P(a_i)\right) \cdot R^P_{i-1} + f_i^N(a_i) \cdot F^N_{i-1} \tag{1}$$

of emails classified as solicited. Emails that are considered as unsolicited by node *i* and therefore are sorted out summarize to the absolute number R_i^N of real negatives and the amount F_i^P of solicited mails falsely sorted out, leading to the total of

$$R_i^N + F_i^P = \left(1 - f_i^N(a_i)\right) \cdot F^N_{i-1} + f_i^P(a_i) \cdot R^P_{i-1}\,. \tag{2}$$

The complete costs C_i resulting from node *i* consist of fixed costs as with $C_i^{fix} : [0,1] \to IR$ and costs per filtered email $c_i^{rel} : [0,1] \to IR$ both restricted to monotonic increasing functions leading to total filter costs of node *i* of

$$C_i = C_i^{fix}(a_i) + c_i^{rel}(a_i) \cdot \left(R^P_{i-1} + F^N_{i-1}\right). \tag{3}$$

The model leads to an optimization task which should both minimize the complete costs caused by all filters as $\tilde{C} = \sum_{i=1}^{n} C_i$ and minimize the rate of false positives $\tilde{f}^P = \frac{1}{R_0^P} \sum_{i=1}^{n} F_i^P$.

Note that when formulating the utility function to be optimized, we distinguish between the costs to filter the emails on the one hand and the rate of false positives on the other hand. Based on past experience, it is comparatively easy to evaluate the damage that spam emails are causing an organization when left unfiltered. Major components of these costs include the time of the users that is needed to identify spam and delete it and potential viruses that may affect an organization's IT. The case of false positives is much more complicated. Depending on the industry – e.g. take a law firm as opposed to a used-book reseller – false positives might incur substantially different costs. Moreover, a single false positive might cause substantial damage in an organization, but many others might not cause any damage at all. Thus, we expect the relationship between the number of false positives and the caused damage to follow some power law distribution that can hardly be quantified. That is why we use a punishment factor in our suggested utility function in the following instead of a monetary quantification of the average expected costs false positives will cause.

This leads to a function $F:(a_1,\ldots,a_n)^T \to IR$ representing an aggregated utility function – or "fitness" function – of the form

$$F(a) = \tilde{C} \cdot P\left(\tilde{f}^P\right) = \left(\sum_{i=1}^{n} C_i\right)^{-1} \left(1 - \frac{1}{R_0^P} \sum_{i=1}^{n} F_i^P\right)^{-\frac{\ln 100}{\ln(1-b)}} \qquad (4)$$

with $a=(a_1,\ldots,a_n)^T$ as the vector of intensity levels, $P:[0,1] \to [0,1]$ as a strictly monotonic decreasing function with $P(0) = 1$ punishing small values of $\tilde{f}^P$ already strongly, and $b \in (0,1)$ as the complete rate of falsely classified wanted emails causing the punishment factor to be $P(b) = 0.01$. Take for example $b = 0.001$ and a rate of false positives of $\tilde{f}^P = 0.002$. These values would already result in a strong punishment of the utility in the fitness function of $(1/100)^2 = 1 \cdot 10^{-4}$. Since the "fitness" function contains the reciprocal of the costs, we believe that the constant b can be used as a suitable and economically still meaningful control parameter for false positives. To minimize $\tilde{C}$ and P the "fitness"-function $F(a)$ has to be maximized.

It is assumed in our model that based on historical observations and the experience of the system administrator, the average stream of incoming emails, the average number of spam emails among the incoming emails and the average structure of the stream of incoming emails in terms of different types of spam emails are all known. But even with this information in hand, the solution of the filtering approach is not trivial, as we will see in the following section.

Solution

$F(a)$ is in general highly non-linear and not globally concave due to its construction, and the relatively weak restrictions regarding the functions f_i^N, f_i^P, C_i and c_i^{rel}. (Gill et al. 1982) state that gradient based local hill-climbing techniques are frequently used to maximize functions like $F(a)$, which span a broad range of function classes with a high risk of *not finding* a *global maximum* but ending in a *local maximum*. Thus, automatically finding a global maximum as a built-in feature of an adequate decision support tool constitutes a challenge. Since we feel that such a decision support tool ought to provide for a solution "at the touch of a button", we suggest a heuristic approach using an evolutionary strategy to provide as good a solution as possible for the wide range of possible occurrences of $F(a)$. Evolutionary strategies are based on biologically motivated reproduction and selection strategies and have proven to be useful for solving a large range of problems which occur with optimization algorithms as described by (Nissen 1997). Evolutionary strategies use a collection of heuristic rules to modify a population of trial solutions in such a way that each generation of trial values tends to be on average better than its predecessor.

A pure evolutionary strategy as e.g. in (Rechenberg 1973) or an improved approach using self-adapting mutation parameters such as the Covariance Matrix Adaptation Evolution Strategy (CMA-ES) algorithm of (Hansen and Ostermeier 2001) do not utilize the potential differentiability of the objective function. However, our objective (or fitness) function is differentiable. Thus, a combination of an evolutionary strategy with the quasi-Newton method (gradient-based approach) may substantially accelerate the solution procedure.

Following the suggested approach in (Sekhon and Mebane 1998), we chose the Broyden-Fletcher-Goldfarb-Shanno (BFGS) method (Broyden 1970; Fletcher 1970; Goldfarb 1970; Shanno 1970) to solve our problem at hand. The BFGS method is a very prominent representative in the class of optimization algorithms that combines these two aspects. It is used to improve local best solutions faster to a nearby local optimum. The BFGS method uses finite differences based on optimal intervals to calculate estimators of local gradients as described in (Gill et al. 1982). Further descriptions of this method can be found in (Avriel 2003; Bonnans et al. 2006; Fletscher 1986; Luenberger and Ye 2008; Nocedal and Whright 2006), while it is applied e.g. in (Csurös and Miklós 2009; Kawai et al. 1993; Richardson et al. 2007).

To optimize $F(a)$ a set of $s \in IN$ vectors $a(i) \in [0,1]^n$ of randomly generated intensity vectors with $a(i) = (a(i)_1, \ldots, a(i)_n)^T$ build a start population $P^0 = \{a(1), a(2), \ldots, a(s)\}$. One iteration step $t \in IN$ of the algorithm is called a generation. Each generation t passes the following steps:

1) Calculation of the fitness $F(a(i))$ of each individual of population P^t

2) Random selection of the mating pool and the parents:

   a) Save the individual with the best fitness into the mating pool

   b) Add randomly s - 1 (with $\frac{s}{2} \in IN$) individuals out of P^t into a mating pool $M^t = \{\tilde{a}(1), \ldots \tilde{a}(s)\}$ with selection probabilities

   $$p(a(i)) = \left(\sum_{i=1}^{s} F(a(i)) \right)^{-1} \cdot F(a(i))$$

   c) Divide mating pool in randomly selected pairs of parents p_x and p_y:
   $M^t = \{a(1)_{px}, a(1)_{py}, \ldots, a(\frac{s}{2})_{px}, a(\frac{s}{2})_{py}\}$

3) Application of a set of operators to the mating pool M^t leads to the transformed mating pool $\tilde{M}^t$. These operators can be categorized in two classes:

   a) Crossover operators exchange component-wise information between two corresponding parent individuals. In this study one out of the non-uniform, the multiple point and the local-minimum crossover operators as described in (Sekhone and Mebane 1998) is chosen with equal probability for each parent pair and each generation.

   b) Mutation operators change randomly one or more components of one or more individuals. The uniform, the boundary and the non-uniform mutation operators as described in (Michalewicz 1993) are integrated in this study's algorithm.

$\tilde{M}^t$ builds the next generation's population by $P^{t+1} = \tilde{M}^t$. The next generation step starts over at 1 using the individuals of population P^{t+1}. The algorithm ends by meeting the termination criterion

$$\left| 1 - \frac{\max\limits_{a^{t+1}(i) \in P^{t+1}} F(a^{t+1}(i))}{\max\limits_{a^{t}(i) \in P^{1}} F(a^{t}(i))} \right| < \varepsilon . \quad (5)$$

Example

In this section, the model will be illustrated by a very simple, though realistic example. Figure 3 shows the scheme of a filtering sequence with three filters, which will be described below. According to a couple of system administrators of our respective universities, the chosen filters as well as their sequence are a typical architectural setup.

Filter 1 represents a server-based blacklist. This filter tries to match the sender-address or sender-server of each incoming email with a central table of sender addresses known to send (exclusively) spam. In case of a positive match the email is classified as spam. The computational costs per filter process for this type of filter are relatively small. Fixed costs to be taken into account for this filter include administrational expenditures for hard- and software and maintenance costs for the blacklist table. Within this example the blacklist filter is parameterized as follows:

- Fixed Costs $\left(C_1^{fix}\right)$: The number of administrator-man-days times costs per man-day multiplied by a weakly concave function (here, we use the square root), that grows with the filtering-intensity a_1. The function is modeled as weakly concave due to the relatively high initial costs for setting up the filter,
- Relative Costs $\left(c_1^{rel}\right)$: All incoming emails $\left(R_0^P + F_0^N\right)$ times processing time per email times costs per server times intensity a_1,
- False Positives $\left(F_1^P\right)$: All incoming solicited emails $\left(R_0^P\right)$ multiplied by the rate of false positives $\left(f_1^P(a_1) = 1 \cdot 10^{-5} \cdot a_1\right)$ as a very small fraction of the intensity a_1,
- False Negatives $\left(F_1^N\right)$: All incoming spam emails $\left(F_0^N\right)$ multiplied by the rate of false negatives $\left(f_1^N(a_1) = 1 - 0.2 \cdot \sqrt{a_1}\right)$ as a weakly concave falling function.

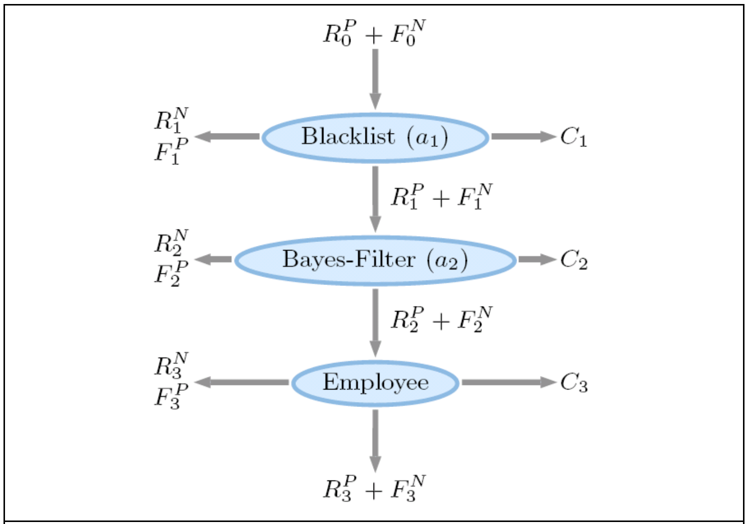

Figure 3. Example as a three-filter-sequence

Filter 2 is implemented as a client-based Bayesian filter as described by (Robinson 2003). Based on conditional probabilities this filter is capable of imitating the individual spam-classifying behavior of each user to set up a database and a set of rules to classify emails as spam or wanted. The processing time for each email is comparatively high for this type of filter and there are initial training costs for every employee. The costs as well as the success rates and failure rates, respectively, of this filter are modeled as follows:

- Fixed Costs $\left(C_2^{fix}\right)$: Training-costs per employee times number of employees times a strongly concave function growing with the intensity a_2 (here we use $a_2^{1/10}$). The function accounts for initial filter training by the employees,
- Relative Costs $\left(c_2^{rel}\right)$: The number of emails $\left(R_1^P + F_1^N\right)$ passing the first filter times processing time per email times costs per server times a strongly concave function of the intensity a_2 (here we use $a_2^{1/4}$),
- False Positives $\left(F_2^P\right)$: The number of wanted mails passing the blacklist $\left(R_1^P\right)$ multiplied by the rate of false positives $\left(f_2^P(a_2) = 10^{-2} \cdot a_2\right)$ as a weakly linear growing function of a_2,
- False Negatives $\left(F_2^N\right)$: The number of spam mails passing the blacklist $\left(F_1^N\right)$ multiplied by the rate of the false negatives $\left(f_2^N(a_2) = 10^{-2 \cdot a_2}\right)$ as a convex falling function of a_2. Based on our experience, this function can reach a relatively high filtering rate.

Filter 3 represents the user which is modeled as the last instance of the filtering sequence. There are no fixed costs connected to this “filter”. We felt that nowadays using emails is very widespread and nearly all email users have already had some experiences with spam emails. Thus, an upfront training is not necessary. Still, fixed costs could easily be included in this example and in the calculation. Costs are caused by the time the employee needs to read, recognize and delete an unwanted email. It is assumed that the employee always works with an intensity of 1 (since $a_3 = 1$, it is omitted in Figure 3), thus he does not delete real positives on purpose. Still, a small percentage of wanted emails are deleted by the employee by mistake. Note that we are aware that a human “filter” will not “work” with the same consistency as the other filters will (Hammond 1996). For the sake of simplification, we abstract from this notion here. Moreover, if the organization reaches sufficient size, averaged values can be used for the calculation. This results in the following parameterization of this filter:

- Relative Costs $\left(c_3^{rel}\right)$: The number of not correctly filtered spam coming from filter 2 $\left(F_2^N\right)$ to arrive at the user times costs per employee times deletion time per email,
- False Positives $\left(F_3^P\right)$: The number of real positives $\left(R_2^P\right)$ from the Bayesian filter multiplied by a very low fixed rate of false positives $\left(f_3^P = 1 \cdot 10^{-5}\right)$, which stands for an average percentage of wanted emails deleted by mistake.

Combining this parameterization, a global sequential model can be set up resulting in a fitness function $F(a_1, a_2)$ which has to be optimized by the heuristic approach described in section 3 (“Solution”). The results lead to recommendations for each filter’s intensity.

The values used in this example are summarized in Table 1 and were cross-checked with a system administrator:

Table 1. Parameters and Values used in Example

Parameters	Values and calculation	Used in
Costs per man day (both administrator and employee)	50,000 euros / 200 working days = 250 euros/working day	C_1^{fix}, C_2^{fix}, c_3^{rel}
Server costs	10,000 euros	c_1^{rel}, c_2^{rel}
Incoming mails per year	20 * Employees * 360 (since spam also arrives on the weekend)	c_1^{rel}
Spam rate (before any filter is applied)	0.7	F_0^N
Time (in years) to process one mail via blacklist filter	$1*10^{-11}$	c_1^{rel}
Maximum number of administrator-man-days per year for the blacklist filter	10	C_1^{fix}
Maximum number of man-days per employee per year for installation. Training of employee, training of filter	0.4	C_2^{fix}
Time (in years) to process one mail via Bayesian filter	$1*10^{-8}$	c_2^{rel}
Average time to manually delete a spam message	12 seconds	c_3^{rel}
Average working hours per day	8 hours	c_3^{rel}

For the case of 110 employees the optimum is reached with the parameters $a_{\max} = (1; 0.306)^T$ and a fit value of $1.964 \cdot 10^{-6}$ in generation 19 of the algorithm described in section 3 (“Solution”). Given the model parameters, further analysis reveals that the effect of changes of the blacklist intensity on $F(a)$ is not as high as changes to the Bayesian filter. In the optimum a fully-featured blacklist filter is suggested ($a_1 = 1$), whereas the Bayesian filter is supposed just to analyze a small part of an email. The optimal intensity of about 0.3 translates into the analysis of e.g. the first 50 characters of the email text.

Further analyses demonstrate the sensitivity of the filters 1 and 2 with regard to the number of employees. As can be seen from Figure 4 a Bayesian filter is already cost-effective starting with one employee in our example.

Furthermore, the model suggests switching a portion of the anti-spam expenditures from the Bayesian filter to set up the fully-featured central blacklist when the critical number of around 100 employees is reached.

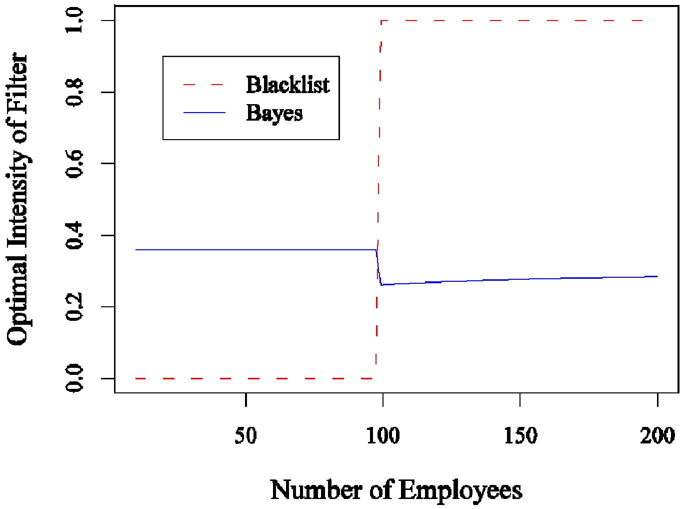

Figure 4. Sensitivity of optimal filter intensities against number of employees

Conclusion

Spam is undoubtedly a serious problem which causes employees to be distracted and to use valuable time to delete spam. As technical methods against this problem exist, institutions and companies have to decide which combination of filters they plan to integrate into their anti-spam strategy. In this paper it has been shown that describing spam filters as nodes in a sequential model may lead to useful recommendations. As a novel approach, we apply a combination of an evolutionary strategy with the quasi-Newton method known as the Broyden-Fletcher-Goldfarb-Shanno (BFGS) method. The BFGS method is used to improve local best solutions faster to a nearby local optimum. In our example we find that the optimal intensity of – and thus the costs associated with – specific filters may depend crucially on the number of employees (here: blacklist filter) due to the regressive effects of fixed costs while others do not (here: Bayesian filter). With the proposed approach it is possible for the first time to evaluate and optimize an overall system of sequential spam filters. Furthermore, the model might be used as a basis for future applications in order to compare different spam filtering architectures.

We are aware that abstracting from interdependencies between different filters in our model and just using the number of spam and solicited emails as input and output of a specific filter node constitute a strong assumption. Therefore the further development of the model is an important object for further research. We are currently working on the integration of a feedback parameter into the model that calibrates the rate of falsely classified spam and the rate of falsely classified solicited emails of specific filter nodes subject to all filter nodes that have been already passed by the stream of emails. Still, based on our discussions with system administrators, the typical order in a filtering architecture is, first, one or several origin-based filters, second, one or several content-based filters and finally the human as the last filter. So it is more the interdependencies between filters caused by changing filter intensities that we want to focus on in the future as opposed to interdependencies resulting from different orders of the filters.

In addition, there are numerous other directions for further research. First, real data should be applied to validate the model. Second, further studies could introduce an implementation of a dynamical change of the filter order, statistical models of the single filter nodes instead of rate-based models and a graphical user interface for the decision maker, that facilitates setting up each filter's parameter. Finally, the determination of the optimal point in time to re-calibrate the proposed model due to changes in the absolute amount and the types of spam emails resulting in changing filter success rates as well as changing cost structures are also subjects of further research.

References

Androutsopoulos, I., Koutsias, J., Chandrinos, K., Paliouras, G., and Spyropoulos, C. 2000. "An evaluation of Naïve Bayesian anti-spam filtering," in *Proceedings of the workshop on machine learning in the new information age*, 11th European conference on machine learning, Barcelona, Spain.

Avriel, M. 2003. *Nonlinear Programming: Analysis and Methods*. Dover Publishing, Mineola.

Bager, J. 2003. "Smarte Spam-Killer - Spam-Blocker für den Desktop," *c't magazin für computertechnik* (17), pp. 138-141.

Bonnans, J., Gilbert, J., Lemaréchal, C., and Sagastizábal, C. 2006. *Numerical optimization, theoretical and numerical aspects*. 2nd edition. Springer, Berlin.

Broyden, C. 1970. "The Convergence of a Class of Double-rank Minimization Algorithms," *Journal of the Institute of Mathematics and Its Applications* (6:1), pp. 76-90.

Cao, Y., Liao, X., and Li, Y. 2004. *An E-mail Filtering Approach Using Neural Network*, Springer, Berlin.

Carpinter, J., and Hunt, R. 2006. "Tightening the net: A review of current and next generation spam filtering tools," *Computers & Security* (25:8), pp. 566-578.

Chou, C., Sinha, A., and Zhao, H. 2010. "Commercial Internet filters: Perils and opportunities," *Decision Support Systems* (48:4), pp. 521-530.

Cranor, L., and LaMacchia, B. 1998. "Spam!" *Communications of the ACM* (41:8), pp. 74-83.

Csurös, M., and Miklós, I. 2009. "Streamlining and large ancestral genomes in Archaea inferred with a phylogenetic birth-and-death model." *Molecular Biology and Evolution* (26:9), pp. 2087-2095.

Delany, S., Cunningham, P., Tsymbal, A., and Coyle, L. 2004. "A case-based technique for tracking concept drift in spam filtering," in *Proceedings of the 24th SGAI international conference on innovative techniques and applications of artificial intelligence*, Cambridge, UK.

Fdez-Riverola, F., Iglesias, E., Diaz, F., Mendez, J., and Corchado, J.M. 2007. "SpamHunting: An instance-based reasoning system for spam labelling and filtering," *Decision Support Systems* (43:3), pp. 722-736.

Fdez-Riverola, F., Iglesias, E., Diaz, F., Mendez, J., and Corchado, J.M. 2007. "Applying lazy learning algorithms to tackle concept drift in spam filtering," *Expert Systems with Applications* (33:1), pp. 36-48.

Fletcher, R. 1970. "A New Approach to Variable Metric Algorithms," *Computer Journal* (13:3), pp. 317-323.

Fletcher, R. 1987. *Practical methods of optimization*, 2nd edition, John Wiley, New York.

Gill, P., Murray, W., and Wright, M. 1982. *Practical Optimization*, 2nd edition, Academic Press, San Diego.

Goldfarb, D. 1970. "A Family of Variable Metric Updates Derived by Variational Means," *Mathematics of Computation* (24), pp. 23-26.

González-Talaván, G. 2006. "A simple, configurable SMTP anti-spam filter: Greylists," *Computers & Security* (25:3), pp. 229-236.

Graham, P. 2004. *Hackers & Painters. Chapter 8: A Plan For Spam*, O'Reilly, Bridgeport, West Virginia.

Green, D., and Swets, J. 1966. *Signal Detection Theory and Psychophysics*, John Wiley, New York.

Hammond, K. 1996. *Human Judgment and Social Policy: Irreducible Uncertainty, Inevitable Error, Unavoidable Injustice*, Oxford University Press, New York.

Hansen, N., and Ostermeier, A. 2001. "Completely Derandomized Self-Adaption in Evolution Strategies," *Evolutionary Computation* (9:2), pp. 159-195.

Kawai, J., Painter, J., and Cohen, M. 1993. "Radioptimization - goal-based rendering." In SIGGRAPH 93 Conference Proceedings, Anaheim, USA, pp. 147-154.

Leiba, B., Borenstein, N. 2004. "A Multifaceted Approach to Spam Reduction", in *Proceedings of the 1st Conference on Email and Anti-Spam* (CEAS), Mountain View, USA.

Loder, T.; Van Alstyne, M., and Wash, R. 2006. "An Economic Response to Unsolicited Communication," *Advances in Economic Analysis & Policy* (6:1), Article 2.

Lueg, C. 2003. "Spam and Anti-Spam Measures: A Look at Potential Impacts," in *Proceedings of the Int. Conference on Informing Science & IT Education*, Pori, Finland.

Lueg, C. 2005. "From spam filtering to information retrieval and back: seeking conceptual foundations for spam filtering," in *Proceedings of 68th Annual Conference of the American Society for Information Science and Technology,* Charlotte, USA, pp. 1313-1314.

Lueg, C., Huang, J., and Twidale, M. 2007. "Mystery Meat Revisited: Spam, Anti-Spam Measures and Digital Redlining," *Webology* (4:1).

Luenberger, D., and Ye, Y. 2008. *Linear and nonlinear programming*, International Series in Operations Research & Management Science, 116, 3rd edition, Springer, New York.

Macmillan, N., and Creelman, C. 2005. *Detection Theory: A User's Guide*, Lawrence Erlbaum Associates, New York.

MessageLabs 2010. *MessageLabs Intelligence: 2009 Annual Security Report*, available at http://www.messagelabs.com/.

Michalewicz, Z. 1993. *Genetic Algorithms + Data Structures = Evolution Programs*, 3rd edition, Springer, Berlin.

Nissen, V. 1997, *Einführung in evolutionäre Algorithmen*, Vieweg, Braunschweig.

Nocedal, J., and Wright, S. 2006. *Numerical Optimization,* 2nd edition, Springer, Berlin.

Rechenberg, I. 1973. *Evolutionstrategie. Optimierung technischer Systeme nach Prinzipien der biologischen Evolution,* Frommann-Holzboog, Stuttgart.

Richardson, M., Dominowska, E., and Ragno, R. 2007. “Predicting clicks: estimating the click-through rate for new ads”. In *Proceedings of the 16th international conference on World Wide Web*, New York, USA, pp. 521-530.

Robinson, G. 2003. “A Statistical Approach to the Spam Problem,” *Linux Journal* (107).

Sekhon, J., and Mebane, W. 1998.“Genetic optimization using derivatives,” *Political Analysis* (7:1), pp. 187-210.

Shanno, D. 1970. “Conditioning of Quasi-Newton Methods for Function Minimization,” *Mathematics of Computation* (24), pp. 647-656.

Van Alstyne, M. 2007. “Curing Spam: Rights, Signals & Screens,” *The Economists' Voice* (4:2), Article 4.