Applied Soft Computing 48 (2016) 111–123

Contents lists available at ScienceDirect

Applied Soft Computing

journal homepage: www.elsevier.com/locate/asoc

A spam filtering multi-objective optimization study covering parsimony maximization and three-way classification

Vitor Basto-Fernandes[a], Iryna Yevseyeva[b], José R. Méndez[c], Jiaqi Zhao[d], Florentino Fdez-Riverola[c,*], Michael T.M. Emmerich[e]

[a] School of Technology and Management, Computer Science and Communications Research Centre, Polytechnic Institute of Leiria, 2411-901 Leiria, Portugal
[b] School of Computer Science and Informatics, Faculty of Technology, De Montfort University, LE1 9BH Leicester, United Kingdom
[c] Informatics Engineering School, University of Vigo, Campus Universitario As Lagoas s/n, 32004 Ourense, Spain
[d] Key Laboratory of Intelligent Perception and Image Understanding of the Ministry of Education, International Research Center for Intelligent Perception and Computation, Xidian University, Xian Shaanxi Province 710071, China
[e] Leiden Institute of Advanced Computer Science, Faculty of Science. Leiden University, 2333-CA Leiden, the Netherlands

ARTICLE INFO

Article history:
Received 29 September 2015
Received in revised form 17 June 2016
Accepted 27 June 2016
Available online 4 July 2016

Keywords:
Spam filtering
Multi-objective optimization
Parsimony
Three-way classification
Rule-based classifiers
SpamAssassin

ABSTRACT

Classifier performance optimization in machine learning can be stated as a multi-objective optimization problem. In this context, recent works have shown the utility of simple evolutionary multi-objective algorithms (NSGA-II, SPEA2) to conveniently optimize the global performance of different anti-spam filters. The present work extends existing contributions in the spam filtering domain by using three novel indicator-based (SMS-EMOA, CH-EMOA) and decomposition-based (MOEA/D) evolutionary multi-objective algorithms. The proposed approaches are used to optimize the performance of a heterogeneous ensemble of classifiers into two different but complementary scenarios: parsimony maximization and e-mail classification under low confidence level. Experimental results using a publicly available standard corpus allowed us to identify interesting conclusions regarding both the utility of rule-based classification filters and the appropriateness of a three-way classification system in the spam filtering domain.

1. Introduction

Nowadays, the use of Internet mailing services has become indispensable in the daily life of millions of users worldwide. Additionally, the combination of e-mail with the latest mobile always-connected smart-phones provides a simple but powerful method to stay in touch with other people and efficiently exchange documents at any time. As a result, both instant messaging (IM) applications and e-mail are commonly used for this purpose. However, a fundamental difference between popular IM applications (including Whatsapp or GTalk) and Internet mailing services is the existence of consent management methods, which can be found only among the former (e.g. blocking users, etc.). This situation has greatly facilitated the use of e-mails as an aggressive/massive advertisement method and virus distribution platform, originating the spam phenomenon.

Since the inception of spam, many companies and research teams have combined their efforts to fight against spam deliveries using different approaches and methods [1]. In this context, and from a scientific perspective, several machine learning (ML) algorithms have been successfully adapted and applied to filter spam messages, mainly including Naïve Bayes (NB) [2], ensemble techniques [3], Support Vector Machines (SVM) [4] and other memory-based systems [5]. Additionally, the computer security industry and the open source community also contributed with effective techniques such as DNS black and white lists [6,7], hashing schemes [8] and the development of SpamAssassin [9], the most popular filtering framework used to combine heterogeneous and complementary anti-spam techniques.

Since its creation, SpamAssassin has been widely used as the base of commercial products and filtering services including McAfee SpamKiller and Symantec Brightmail [10]. It allows system administrators to define specific filters using *ad hoc* rules. Each rule contains a logical expression (used as a trigger) and defines its associated score. Every time an e-mail is received for evaluation, SpamAssassin finds all the rules matching the target message

[*] Corresponding author at: ESEI: Escuela Superior de Ingeniería Informática Edificio Politécnico, Campus Universitario As Lagoas s/n, 32004, Ourense, Spain.
E-mail addresses: vitor.basto.fernandes@ipleiria.pt (V. Basto-Fernandes), iryna.yevseyeva@dmu.ac.uk (I. Yevseyeva), moncho.mendez@uvigo.es (J.R. Méndez), jiaqizhao@stu.xidian.edu.cn (J. Zhao), riverola@uvigo.es (F. Fdez-Riverola), emmerich@liacs.nl (M. T.M. Emmerich).

http://dx.doi.org/10.1016/j.asoc.2016.06.043
1568-4946/© 2016 Elsevier B.V. All rights reserved.

and computes the sum of their scores. This accumulative value is then compared with a configurable threshold (*required score*) to finally classify the new incoming message as spam or legitimate (also known as *ham*).

In order to define accurate anti-spam filters, the SpamAssassin framework provides implementations of several techniques including regular expressions, DNS black and white lists, Distributed Checksum Clearinghouses [11], Naïve Bayes [12,13], Sender Policy Framework [14], Hashcash [15], DomainKeys Identified Mail [16], language guessing [17], as well as several extra protocol error checks. Additionally, SpamAssassin allows the use of user-defined plugins to further extend the number of available techniques that compose a given filter.

Given the configurable structure of the SpamAssassin framework, and taking into consideration that the final accuracy of each user-defined filter strongly depends on the diversity of the underlying classifiers, the optimization of rule weights and other parameters governing the primary rule-based filtering process is still a challenge. In such a situation, initial approaches for the optimization of rule-based filters have been formulated as a single objective problem, where a general performance index (e.g., number of errors, kappa index or f-score, or Total Cost Ratio) is commonly used [18]. However, a more intuitive formulation of this problem involves several objectives. In fact, at least two complementary indexes should be simultaneously considered for minimization in the development of novel accurate anti-spam filters: (*i*) number of false negative (FN) errors (i.e., spam messages classified as legitimate) and (*ii*) number of false positives (FP) errors (i.e., legitimate messages classified as spam). Nevertheless, these objectives are in conflict, since minimizing the number of FP errors can be done only at the expense of increasing the number of spam messages going into e-mail boxes, and vice versa.

Single objective optimization approaches (also known as ‘a priori’ methods) require that sufficient preference information is expressed (blindly) before solution set is computed (i.e. assigning weights for the target objectives to aggregate objectives within a single objective which is optimized subsequently). In contrast, multi-objective optimization (‘a posteriori’) methods provide insights of conflicts between the objectives, i.e. at which extend one objective can be improved at the cost of other(s). Thus, user can select the resulting solution that best fits his/her preferences. In this context, some initial approaches [19,20,10] have evaluated the suitability of applying different multi-objective evolutionary algorithms (MOEA) in the spam filtering domain for optimizing both FN and FP errors at the same time. However, in the aforementioned studies only classical MOEA techniques (NSGA-II and SPEA2) were applied, and questions such as how to better adapt these algorithms using domain specific knowledge and how to consider other objective functions remained unanswered.

In this line, we carried out a preliminary study about the performance of several MOEA approaches to solve different optimization questions [21]. An extended version of our preliminary work, including only the study on parsimony but additional benchmarks, has very recently been published in [22]. The problem of three-way classification and the postprocessing of results on SPAM filters were however not addressed. In detail, this study included the spam filtering problem as a part of the MOEA benchmarking protocol with the goal of showing the insights of conflicts between those objectives to be minimized. However, our past work did not contribute a method to accurately evaluate the structure of the decision space (i.e., a detailed analysis of the relevance of each rule), which is essential for administrators to maintain (and continuously improve) filtering services.

Moreover, in order to fight against spam in environments where the cost associated to misclassification errors is high, the three-way classification scheme [23–25] emerged as a reliable way of mitigating information loss and security risks. Under this scheme, classifiers can avoid providing a solution in case there are not enough evidences to assign target instances to one of the two available classes (i.e., spam or legitimate). In such a situation, these messages are labelled as ‘suspicious’, ‘doubtful’ or ‘borderline’, being the subject of a further examination manually carried out by the final user. In this context, to increase security while revising suspicious e-mails, images, links and dangerous attachments should not be automatically loaded. As long as suspicious e-mails do not count as errors but are classified at the expense of increasing user efforts, the amount of messages labelled in this way should be also minimized (i.e., if all the messages belonging to a given corpus are labelled as ‘suspicious’, the number of misclassifications will be zero). Complementarily, the appropriateness of using a three-way classification scheme was also suggested as future work in our preliminary study [21].

In the present work, we complement previous findings by using three modern plus two classic MOEA approaches in two different ways: (*i*) by studying the structure of the decision space in the optimization of traditional binary classification processes (i.e., minimizing the amount of necessary rules and the number of FP and FN errors) and (*ii*) evaluating the suitability of three-way classification schemes to accurately filter spam contents. In the former case, we take advantage of the first optimization objective (parsimony) to specifically assess the contribution of each rule when generating a correct classification. In the second case, we carry out a performance study about the minimization of FP and FN errors when working with a three-way classification filter. These analyses have been implemented as two different optimization scenarios, making use of a well-known publicly available corpus.

While this section has introduced the motivation for this work, the rest of the paper is organized as follows: Section 2 presents the problem formulation, explains how to optimize ML classifiers with evolutionary algorithms (EAs) and summarizes previous works in anti-spam filter optimization using MOEA. Section 3 introduces the two case studies, defines the benchmarking protocol, establishes the performance metrics to be used and presents and discusses relevant issues regarding each case study. Finally, Section 4 provides conclusions and identifies future research work.

2. Materials and methods

In spite of the fast progress in computer technology and the constant increase of computational power, performing exhaustive searches in large continuous and combinatorial spaces is still challenging. In this context, the remarkable popularity of EAs over other optimization techniques is mainly motivated by their ability to search these spaces and find approximate (near) optimal solutions [26]. In the particular domain of multi-objective optimization, EAs stand for well-established computational methods where the population-based approach makes them suitable to search for approximation sets to the efficient set.

In this way, EAs were found to be particularly useful for dealing with multi-objective problems characterized by several conflicting goals, for which not simply a single optimum solution, but a set of Pareto optimal or non-dominated solutions need to be obtained. Together, these solutions represent the trade-offs between the existing objectives, being optimal in the sense of Pareto dominance. In such a situation, a Pareto optimal solution can only be improved in one objective at the expense of loss in other(s). As long as a population of possible solutions is used in parallel to solve these problems, the search is directed not a single optimum but towards multiple Pareto optimal solutions, which is the case of MOEAs (also known as Evolutionary Multi-objective Optimization Algorithms, EMOAs).

Table 1
Confusion matrix of traditional (binary) classifiers.

		True class	
		P	N
Predicted class	P	True Positives	False Positives
	N	False Negatives	True Negatives

Table 2
Confusion matrix of three-way classifiers.

		True class	
		P	N
Predicted class	P	True Positives	False Positives
	N	False Negatives	True Negatives
	?	Further Exam	

The constant development of novel MOEAs while trying to achieve better performance with respect to the quality of the obtained set of solutions (according to both convergence and coverage of the Pareto front approximation) led to the existence of several generations of MOEAs [27] been created. The theoretical foundations of these approaches are thoroughly discussed in [28] while more recent approaches can be found in the works of Laumanns [29] and Auger et al. [30]. Additionally, a large number of real-world applications are also discussed in the work of Coello [27].

2.1. Problem formulation: spam filtering domain perspective

As previously discussed, in the context of traditional binary classifiers, misclassifications are commonly grouped into FP and FN errors. In order to correctly apply EA to optimize anti-spam filters, a normalized counting of the false negative and false positive occurrences is adopted. These measures are called as false negative rate (FNR) and false positive rate (FPR), respectively. Expression (1) shows how to compute their corresponding values.

$$\mathrm{FNR} = \frac{\mathrm{FN}}{\text{Total Number of Spam Messages}} \quad \mathrm{FPR} = \frac{\mathrm{FP}}{\text{Total Number of Legitimate Messages}} \quad (1)$$

Additionally, when working with traditional ML classifiers, their hits can be separated into true positives (TP) and true negatives (TN) classes, depending on whether the target message was really spam or legitimate, respectively. In this context, the relationship between the true labels and the predicted ones can therefore be presented in a two-by-two straightforward confusion matrix as shown in Table 1.

Taking into consideration the values that are part of the confusion matrix, FNR and FPR can be easily computed, as shown in Expression (2).

$$FNR = \frac{FN}{TP + FN} \quad FPR = \frac{FP}{TN + FP} \quad (2)$$

However, if we consider only rigid binary classifiers, for which every new instance is simply categorized as positive or negative, it may result in high number of misclassifications leading to high costs. To reduce these errors, the final user of an anti-spam filter could help in the classification task with the goal of improving the accuracy of the filter and reducing its associated cost. In this context, the use of a three-way classification scheme allowed us to take advantage of final users both to improve the classification accuracy of the filter and to maximize the user experience. In such a situation, the initial confusion matrix (introduced in Table 1) changes to become a three-by-two matrix, as shown in Table 2.

2.2. Optimizing ML classifiers with EAs

Traditional ML classification tasks involving parameter optimization and model selection can be successfully reformulated as multi-objective optimization problems. In fact, they usually require achieving improvements on several conflicting goals, such as recall/sensibility, precision/specificity and classifier complexity [21], simultaneously. Apart from this classical perspective, many other examples of multi-objective optimization of ML classifiers can also be considered, such as the trade-off between learning new information and/or forgetting old one, or between learning as many details as possible and generalizing the model to its maximum in pattern recognition [31]. However, it is only relatively recently that the design of ML systems was conceived from a multi-objective point of view, considering simultaneous optimization of multiple conflicting objective instead of combining them into a single objective. Due to large combinatorial space of such problems, solving them with exact algorithms is not possible in most of cases, hence they are often solved using evolutionary approaches, MOEAs in particular.

In this regard, a usual way to measure the performance of different ML classifiers (or different configurations of the same model) is through the Receiver Operating Characteristic (ROC) curve, which conveniently summarizes the classifier performance when varying discriminative thresholds for two-class classification problems. ROC Convex Hull (ROCCH) is currently being widely used by the scientific community, being able to represent the convex hull area of a set of points, each of which stands for an optimal classifier [21]. Maximizing ROCCH leads to finding the group of classifiers that together provides the best range of optimal classifiers.

Even though the concepts of ROC Convex Hull and the Pareto front were reported to be similar (leading to the application of EMOA for approximating ROCCH [32]), a specific and important property of ROCCH makes it more valuable than the Pareto front. When using ROCCH, any two classifiers belonging to the Convex Hull can be joined using a line in which a new virtual classifier is represented as a point with its corresponding performance [33]. This property can be straightforwardly used to save computational resources [32].

When dealing with a multi-objective optimization problem, there are typically $m$objective functions, $f = (f_1, f_2, \ldots, f_m)$, which are simultaneously optimized (e.g., minimized) so that each f_k, $k \in \left\{1, \ldots, m\right\}$ stands for a real-valued function evaluated in the multi-objective space. Additionally, complementary constraints of equality or inequality could be imposed on the decision variables. The result of a multi-objective optimization approximates the set of Pareto optimal solutions that corresponds to the set of non-dominated solutions found, based on the evaluation of the Pareto dominance relation. In this context, a$y$the Pareto solution dominates any other y' alternative $(y, y' \in R^m)$ if y is better on at least one objective, and is not worse in the remaining cases. The selection of a single solution among the set of Pareto optimal solutions is usually done by decision maker(s). As this is not an easy task, different decision-aiding tools able to take into account the preferences of decision maker(s) are used to help judge.

The Pareto fronts achieved by MOEAs are usually approximated by a finite number of points. In fact, the hypervolume indicator (i.e., a bounded size set of points that jointly dominates a maximal part of the objective space relative to a reference point) or the Convex Hull (which dominates a large part of this space) are commonly used for this purpose. The latter is argued to be more appropriate when it comes to ROC curves approximation [32].

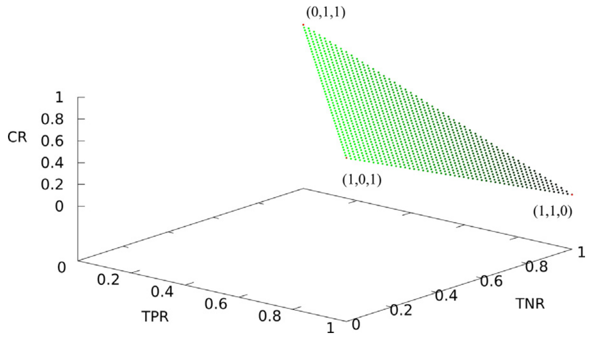


Fig. 1. 3D ROC surface (TPR, TNR, CR) showing the possible situations in a three-way classification scenario.

In a three-way classification scenario, apart from the recommendable maximization of TPR and TNR, the third obvious objective is to maximize the number of instances labeled as spam or legitimate by the filter, namely, the classified instances rate (CR) or coverage. Considering a graphical 3D plot of all these variables (see Fig. 1), the point (0,0,1) represents the situation in which all the instances are incorrectly classified by the filter. In the same line, the point (1,0,1) represents the situation where all the instances are classified as positive by the filter. And the point (0,1,1) corresponds to the case in which all the instances are classified as negative by the filter. Conversely, the point (1,1,0) represents the situation in which all the instances remain unclassified and therefore there are no misclassifications. Finally, the point (1,1,1) stands for the case with all the instances correctly classified, which corresponds to a perfect classifier. A perfect classifier does not usually exist, but the challenge here is to find some classifiers as close to the perfect one as possible, and explore the trade-offs among them.

In Fig. 1, the surface $x+y+z=2$ (where x axis represents TPR, y stands for TNR and z symbolizes CR) represents a situation in which all the instances categorized by the filter are randomly guessed. Establishing the ratio of positive and negative instances as $p(p)$ and $p(n)=1-p(p)$, respectively, and also considering that 80 percent of the instances are processed by an expert, and the remaining 20 percent are predicted by a filter, the point to which the classifier is mapped to is $(0.2\times p(p)+0.8, 0.2\times p(n)+0.8, 0.20)$, as shown in Expression (3).

$$TPR = 0.2\times p(p)+0.8 \tag{2}$$

$$TNR = 0.2\times p(n)+0.8 \tag{3}$$

$$CR = 0.2$$

Since $p(p)+p(n)=1$, it is easy to find out that the surface of $TPR+TNR+CR=2$ graphically represents the set of worst possible solutions. Consequently, maximizing ROC Convex Hull in a three-way classification scenario consists of maximizing the volume above the surface $x+y+z=2$, where the points (1,1,0), (0,1,1,) and (1,0,1) are treated as fixed points on a three-way classifier ROC plot.

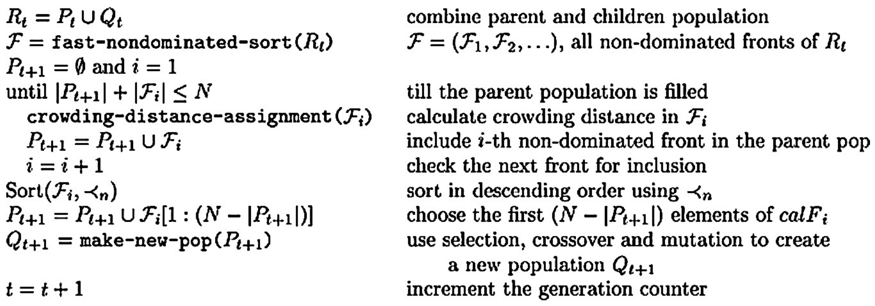
2.3. Relevant advances on multi-objective optimization methods

Considering a general (not problem-specific) EMOA, the selection operator chosen has a huge influence over its efficiency. In each iteration, this operator picks out individuals to be passed to the next generation and hence, parents in the mating phase. In this context, the evolution of EMOAs has been widely influenced by all the research done in the scope of the selection operators.

Pareto-based EMOAs, such as Non-dominated Sorting Genetic Algorithm (NSGA) [34] and Multi-Objective Genetic Algorithm (MOGA) [35] were the first approaches using Pareto non-dominance in the ranking and selection of individuals. The next is the group of elitist EMOAs, which includes NSGA-II [36] and Strength Pareto Evolutionary Algorithm (SPEA) [37], is characterized by selecting non-dominated solutions and allowing their preservation with respect to earlier generations. Another different approach was adopted in the design of region-based EMOAs, such as the Pareto Envelope-based Selection Algorithm (PESA-II) [38], with its focus centered in the competition of regions of the objective space instead of rivalry of individuals. This strategy is similar to the concept of niching [39] with a restricted amount of individuals being selected from each niche in the objective space (note that niching in the decision space is also common [40]).

Later, the development of performance assessment measures (indicators) revealed the possibility of using EMOAs performance indicators directly in the selection operator, which gave birth to indicator-based EMOAs, such as the Indicator-Based Evolutionary Algorithm (IBEA) [41] and the Multi-objective Selection Based on Dominated Hypervolume (SMS-EMOA) [42,43]. In general, any performance indicator may be used in IBEA, but binary indicators preserving Pareto non-dominance are commonly adopted. For instance, hypervolume and epsilon indicators satisfy the desired properties (e.g., compliance with the Pareto dominance principle and monotonicity) and are commonly suggested as the default choice for the selection operator to be based on. Additionally, it was also reported that the hypervolume indicator selects extreme points and concentrates search around the knee region of the Pareto front [42].

NSGA-II stems from its predecessor NSGA and adopts the same non-dominated sorting procedure for allocating all solutions to classes with respect to their non-dominating rank [44]. In order to address the mating selection of parents that will participate in offspring production and environmental selection (from both parents and offspring), individuals from the fronts with best (first) ranks are preferred. In particular, parents are randomly chosen from the population and compared in a binary tournament. If two individuals appear to be from fronts with different ranks, then the winner to enter matting pool of individuals is the one from the non-dominated front with better rank. Moreover, if two individ-



```
$R_t = P_t \cup Q_t$                                 combine parent and children population
$\mathcal{F}$ = fast-nondominated-sort($R_t$)        $\mathcal{F} = (\mathcal{F}_1, \mathcal{F}_2, \ldots)$, all non-dominated fronts of $R_t$
$P_{t+1} = \emptyset$ and $i = 1$
until $|P_{t+1}| + |\mathcal{F}_i| \leq N$           till the parent population is filled
  crowding-distance-assignment($\mathcal{F}_i$)      calculate crowding distance in $\mathcal{F}_i$
  $P_{t+1} = P_{t+1} \cup \mathcal{F}_i$             include $i$-th non-dominated front in the parent pop
  $i = i + 1$                                        check the next front for inclusion
Sort($\mathcal{F}_i, \prec_n$)                       sort in descending order using $\prec_n$
$P_{t+1} = P_{t+1} \cup \mathcal{F}_i[1 : (N - |P_{t+1}|)]$   choose the first $(N - |P_{t+1}|)$ elements of $calF_i$
$Q_{t+1}$ = make-new-pop($P_{t+1}$)                  use selection, crossover and mutation to create
                                                       a new population $Q_{t+1}$
$t = t + 1$                                          increment the generation counter
```

Fig. 2. General overview of NSGA-II algorithm.

```
$k = 0$
repeat
  $k \leftarrow k + 1$
  $R.k \leftarrow$ non-dominated solutions in $P$
  $P \leftarrow P \setminus R.k$
until $P = \emptyset$
return $R_1 \ldots R_k, k$
```

$R_1, \ldots, R_i, \ldots, R_k$ denote partitions of rank $i = 1, \ldots, k$

Fig. 3. Details of non-dominated sorting.

```
crowding-distance-assignment ($\mathcal{I}$)
$l = |\mathcal{I}|$                                          number of solutions in $\mathcal{I}$
for each $i$, set $\mathcal{I}[i]_{distance} = 0$            initialize distance
for each objective $m$
  $\mathcal{I} = \text{sort}(\mathcal{I}, m)$                sort using each objective value
  $\mathcal{I}[1]_{distance} = \mathcal{I}[l]_{distance} = \infty$   so that boundary points are always selected
  for $i = 2$ to $(l - 1)$                                   for all other points
    $\mathcal{I}[i]_{distance} = \mathcal{I}[i]_{distance} + (\mathcal{I}[i+1].m - \mathcal{I}[i-1].m)$
```

Fig. 4. Details of crowding distance sorting.

```
$P_0 \leftarrow$ initialize() {Initialize random start population of $\mu$ individuals}
$t \leftarrow 0$
repeat
  $x_{t+1} \leftarrow$ generate($P_t$) {Generate one offspring by variation operators}
  $P_{t+1} \leftarrow$ replace$_{\Delta S}$($P_t \cup \{x_{t+1}\}$) {Select $\mu$ individuals for the new population}
  $t \leftarrow t + 1$
until stop criterium reached
```

Fig. 5. General overview of SMS-EMOA algorithm.

```
$\{R_1, .., R_\ell\} \leftarrow$ non-dominated-sort($Q$) {all $\ell$ partitions of $Q$ in increasing order}
for all $x \in R_\ell$ do
  $\Delta_{\mathcal{S}}(x, R_\ell) \leftarrow \mathcal{S}(R_\ell) - \mathcal{S}(R_\ell \setminus \{x\})$
end for
$x \leftarrow$ arg min$_{x \in R_\ell}[\Delta_{\mathcal{S}}(x, R_\ell)]$ {detect element of $R_\ell$ with lowest $\Delta_{\mathcal{S}}(x, R_\ell)$}
$Q' \leftarrow Q \setminus \{x\}$
return $Q'$
```

Fig. 6. Details of the *replace* procedure belonging to the SMS-EMOA algorithm.

uals appear to be from the same front, the comparison is done based on their crowding distance (the one with the largest crowding distance is preferred). Similarly, to fill the next population of individuals from the union of parents and offspring populations, individuals from fronts having better ranks are selected first. When there are more available individuals than needed from the last front, those with the largest crowding distance (having the largest distance to the nearest neighbors) are selected. However, the crowding distance mechanism of diversity preservation works poorly when there are more than three objectives. The general framework of NSGA-II is described in the algorithm showed in Fig. 2 and details of non-dominated sorting and crowding distance sorting are provided in the algorithms showed in Figs. 3 and 4, respectively.

Many other methods were also developed to improve the performance of NCSGA-II and to address diversity preservation in the selection operator. In this line, SMS-EMOA, one of the earliest hypervolume-based methods, performs a non-dominated sorting of population to build a single offspring during an initial stage. Then, the offspring is ranked against the already sorted population and an individual from the last non-dominated front with the smallest hypervolume contribution is removed. Fig. 5 describes the general framework of the SMS-EMOA approach whilst Fig. 6 introduces the details of its *replace* procedure.

The success of SMS-EMOA motivated developers to create the steady state version of NSGA-II [45], in which only a single offspring is created in each generation. Therefore, only the worst individual is removed from the population at each iteration of the algorithm. Indeed, the steady state version of NSGA-II performs better than the original one [36] at the expense of higher computational costs.

A completely different approach is used in Multi-Objective Evolutionary Algorithm Based on Decomposition (MOEA/D). This algorithm divides a multi-objective problem into different simpler optimization sub-problems (e.g., defined as scalar aggregation functions) and optimize them simultaneously by only taking into account the information on neighboring sub-problems [46]. To do so, MOEA/D uses an array of objectives and a set of weight vectors to search for different points on the Pareto front. The authors of MOEA/D suggested decomposing the multi-objective problem into as many sub-problems as the number of objectives. For each sub-problem, the corresponding scalarization problem should be solved by using a mathematical programming method (e.g., weighted sum, Tchebycheff, boundary intersection, etc.). By following this approach, the first approximations of the Pareto front can be obtained and, consequently, a selection of the most convenient parents to produce the next generation of individuals can be made. Moreover, the mating pool of several closest neighbors is kept for producing offspring. Finally, all parents are substituted in new MOEA/D generations, but non-dominating solutions with respect to previous generations are preserved.

The Convex Hull-based Multi-objective Optimization Genetic Programming algorithm (CH-MOGP) [32] is an indicator-based MOEA that seeks to maximize the area under the ROC Convex Hull. CH-MOGP was developed under the hypothesis that AUCH is a better indicator to guide the search when compared to the hypervolume for optimizing classifier performance. In fact, two hard classifiers can be successfully combined into a new classifier with its FP and FN rates situated on the line segment connecting the two hard classifiers in the ROC space. Hence, instead of considering only the hypervolume of points representing the hard classifiers included in the population, the hypervolume covered by any linear combination of the points must also be considered. It can be computed as the area under the convex hull of the population augmented by the three additional points (1,0), (0,1) and (1,1), where *x* and *y* axes represent the FN and FP rates, respectively.

The Convex Hull Evolutionary Algorithm (CH-EMOA) [21] can be used as an adaptation of CH-MOGP for parametric search spaces (with bit strings and continuous vectors). It can address different kinds of classification problems using the same selection scheme and using a representation of the solutions in the form of a parsed tree. In brief, CH-EMOA follows the same principles as SMS-EMOA but with three important differences. First of all, instead of using non-dominated sorting to determine different ranks in the first selection step, Jarvis' algorithm for computing the convex hull is applied repeatedly on the population augmented by the point (1,1). In each step, those points lying on the convex hull are determined and removed from the set. Secondly, if the first ranking does not yield a decision about whether or not a point is selected, those points that most contribute to the AUCH are selected. In a $(\mu + 1) -$ *selection* this is accomplished with an exact scheme, otherwise a greedy scheme is applied to keep computational efforts low. In order to compute the contribution of a single point to the AUCH, its two neighboring points on the convex hull are retrieved and the area of a triangle spanned by these three points is computed. Finally, redundant points that are already present in the archive are always removed with priority. Fig. 7a shows the AUCH indicator while Fig. 7b presents the selection scheme.

2.4. Available datasets for anti-spam research

With the goal of boosting the design of novel and accurate filters and the execution of new anti-spam experiments, several companies in conjunction with the scientific community have publicly shared their e-mail datasets (corpus). All these available corpora

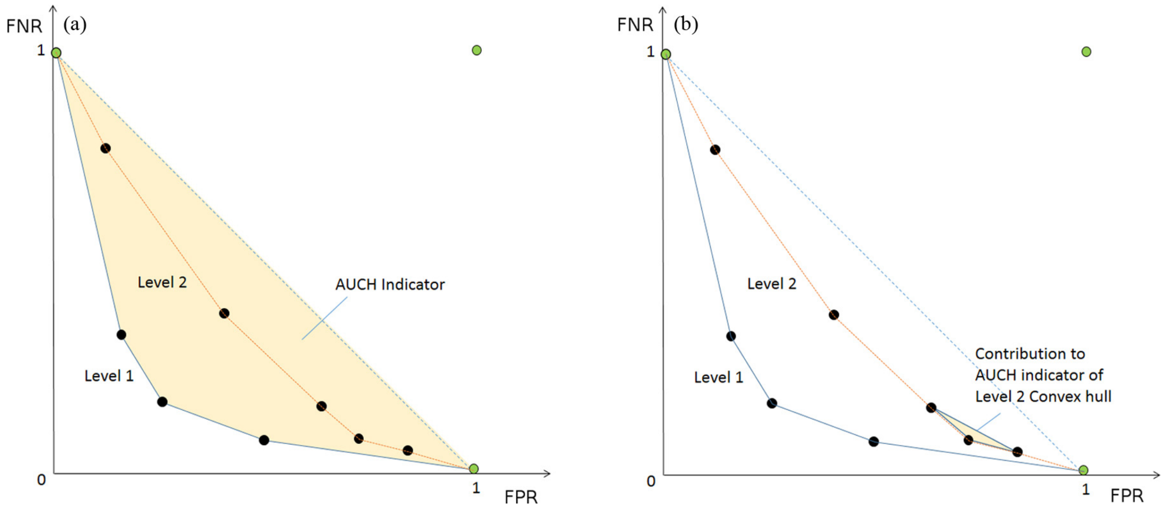


Fig. 7. a) Area under the convex hull (AUCH) indicator for an example set; b) CH-EMOA selection scheme.

Table 3
Publicly available datasets for anti-spam research.

Dataset name	Size	%Ha m	%Spa m	URL
SpamAssassin	9349	74.49	25,51	http://www.spamassassin.org
Bruce Guenter	171000	0	100	http://untroubled.org/spam/
CSDMC2010	4327	68.1	31.9	http://csmining.org/index.php/spam-email-datasets-.html
TREC_Spam_2005	92189	43	57	http://trec.nist.gov/data/spam.html
TREC_Spam_2006	37822	35	65	http://trec.nist.gov/data/spam.html
TREC_Spam_2007	75419	33.5	66.5	http://trec.nist.gov/data/spam.html
PRA_JMLR_2004	17	0	100	http://prag.diee.unica.it/public/datasets/spam/spamArchiveFull/
PRA_JMLR_2005	142876	0	100	http://prag.diee.unica.it/public/datasets/spam/spamArchiveFull/
PRA_JMLR_2006	25522	0	100	http://prag.diee.unica.it/public/datasets/spam/spamArchiveFull/
EnromCorpus	52076	37	63	http://prag.diee.unica.it/public/datasets/spam/Enron/

can be used for testing purposes, enabling results to be compared between different research works. Table 3 compiles the most popular datasets that can be freely downloaded from the Internet.

3. Experimental study

In order to correctly study the advantages of applying three novel indicator and decomposition-based evolutionary multi-objective algorithms (i.e., SMS-EMOA, CH-EMOA, and MOEA/D) to the field of spam filtering, this section (*i*) defines two different but complementary scenarios: parsimony maximization and e-mail classification with three-way classifiers or under low confidence level, (*ii*) establishes the benchmarking protocol and documents the parameter setup, and (*iii*) identifies and comments the performance measures used in the experiments.

3.1. Instantiation of problem domain and definition of scenarios

In our study, spam filtering is formulated as a multi-objective optimization problem characterized by real-valued objective functions $f(y)$ with values in the [0,1] interval, such as FPR and FNR. Minimization is assumed for all the objectives, which are evaluated with individuals collected from decision space as a vector of decision variables $y_1, y_2, \ldots, y_n$ where $i \in \left\{1, \ldots, n\right\}$ represents the score vector of the filter. In the case of well-known rule-based anti-spam filters (such as those generated by SpamAssassin), the vector of scores is represented by an array of decision variables, y of length, n (i.e., the total number of filtering rules), where each variable y_i corresponds to the score of one rule. The individuals that are part of the initial population are randomly generated with scores in the $[-5, 5]$ real variable range. Additionally, new individuals are further generated by using of the crossover and mutation operators in the same range. These settings follow both SpamAssassin anti-spam filter guidelines and configuration included in Debian GNU/Linux OS (Operating System).

As previously commented, with the specific goal of complementing our previous study [21], we have implemented two different but complementary optimization scenarios: (*i*) the minimization of the filter operation complexity through parsimony maximization and (*ii*) the use of three-way classifiers able to label a given message as spam, legitimate or ‘borderline’ (in case of low confidence).

In the first scenario (parsimony maximization), we use a tri-objective binary-real representation formulation, where a typical user would wish to minimize FNR, FPR and Complexity Rate (CR) or percentage of active rules with a score different from zero. In this problem formulation all the objective values fall in the [0,1] range, but a binary decision variable is added to the representation of the decision vector. The purpose is to have one bit for representing each available rule, allowing its activation or deactivation depending on its associated relevance in the whole classification process. This problem formulation, initially introduced in [21], is maintained here with two main purposes: (*i*) assuring accurate performance comparisons with previous approaches and (*ii*) enabling a relevance analysis to consider the different types of rules. In detail, our previous work was focused in providing an appropriate benchmark procedure to compare several MOEA approaches aimed to solve different optimization problems. In contrast, the present work takes advantage of the same evaluation scenario to assess the contribution of each rule for accurately classifying incoming messages.

The second scenario stands for a tri-objective real-valued representation for three-way classification. As in the previous case, a real decision vector is used for representing the scores of all the available anti-spam rules. Additionally, two real variables in the interval [0,1] are used with the goal of representing the two threshold values used for establishing the bounds that define 'unclassified' e-mails. An additional constraint is introduced here in order to guarantee that the lower bound value is smaller than the upper one. By following this problem formulation, three labels are required for classifying e-mails: *legitimate*, *spam* and *unclassified*. Therefore, whenever the target message achieves a score below the lower bound threshold, it is classified as *legitimate*. Conversely, if the e-mail score is above the upper bound threshold, the message is classified as *spam*. Otherwise, if the score is located inside the interval, the message is considered for a further exam (*unclassified*). Under this situation, the rationale is that it is better to leave an e-mail unclassified than to provide a wrong classification.

In our second scenario, the three objectives are the minimization of FNR, FPR and the unclassified ratio of e-mails (UR). This third objective also falls in the range of [0,1], as shown in Expression (4).

$$\mathrm{UR} = \frac{\text{Number of Unclassified Messages}}{\text{Total Number of Messages in the Dataset}} \tag{4}$$

3.2. Benchmarking protocol and parameter setup

In order to guarantee the reproducibility of our experiments, this subsection introduces a straightforward description of all the configuration details needed to run our experimental tests including the target filter definition, the available datasets, and different parameter details regarding the configuration of the executed algorithms.

With reference to the target filter to be optimized, we selected the default and standard spam filter configuration included in the Debian GNU/Linux Squeeze distribution running SpamAssassin 3.3.1 [9]. This decision was mainly motivated by the fact that SpamAssassin (*i*) is the de facto standard in the spam filtering industry, (*ii*) it is publicly available for research projects and novel developments, and (*iii*) it achieves a good performance when considering the effectiveness of the classifications carried out. Regarding its initial configuration, we kept the default value for the *required_score* threshold in all the experiments (i.e., *required_score* = 5). The range of scores during the optimization process was limited to the [−5, 5] interval. Although the default filter definition contains a collection of 2440 different rules, the great majority does not fit any incoming e-mails. In fact, only 330 rules match some e-mails, so only those rules were finally selected to be part of our multi-objective optimization process.

From all the available alternatives shown in Table 3, we finally selected the well-known SpamAssasin corpus [47] in order to run our experimental testbed. Our selection guarantees a medium-sized corpus (containing 9349 e-mails) providing both legitimate and spam messages (6951 legitimate *vs* 2398 spam) characterized by a legitimate/spam ratio very similar to the proportion of current e-mail in-boxes. Moreover, SpamAssassin corpus has been distributed in the same raw format as messages were transmitted through Internet (RFC 5322 format [48]). Hence, using the *spamc* and *spamd* tools included in the SpamAssassin package [9], we can easily compute those matching rules for each available message (*spamc* −y < file.rfc5322.eml). The output of the previous command is saved to a file, which is later used to improve the evaluation speed for each configuration.

As previously stated, in this work we apply three novel indicators and decomposition-based MOEAs for the three-objective optimization of two untested scenarios in the spam filtering domain. In detail, the tested algorithms are the Convex-Hull Evolutionary Multi-objective Optimization Algorithm (CH-EMOA), the Multi-objective Selection Based on Dominated Hypervolume (SMS-EMOA) and the Decomposition-based Multi-objective Evolutionary Algorithm (MOEA/D). Additionally, we also compare the results achieved by these novel approaches with those obtained by classic MOEAs (i.e., NSGA-II and SPEA2) acting as a baseline. All the experiments were performed with jMetal 4.3 [49], an optimization framework for the development of multi-objective metaheuristics in Java.

Similar experiment configurations were adopted for both scenarios (parsimony maximization and three-way classification). In detail, the RealBinary encoding formulation was implemented using the jMetal RealBinary encoding scheme, where the chromosome is constituted by an array of real values in the [−5, 5] interval, plus a bit string. Each filtering rule was associated with a real score value within this interval, plus one bit in the chromosome. In this context, if the ith bit in the chromosome is 0, then the *i*th rule is ignored, otherwise the rule is active, being considered by the filter with the ith bit value corresponding to its real score (rule weight). Motivated by the number of rules comprising our filter (the standard spam filter configuration included in SpamAssassin 3.3.1), the length of the chromosome used in our experiments was $330 \times 2 = 660$.

For all the experiments, we established a maximum number of 25,000 function evaluations. Both the SBX single point crossover and bit flip mutation operators were applied to manipulate binary data in the tri-objective binary-real representation experiments. Additionally, the SBX crossover and polynomial mutation operators were used to manipulate real data belonging to both problem representations (i.e., tri-objective real-valued and tri-objective binary-real). For binary variables, a bit flip mutation was used with a probability of $1/n$ for each single bit. The crossover probability was $p_c = 0.9$ and the mutation probability was $p_m = 1/n$, being n the number of available filtering rules. The population size was set to 100 individuals for all the algorithms. The archive size was established to 100 for the SPEA2 approach, while the offset was assigned a value of 100 for SMS-EMOA and CH-EMOA. All the algorithms were executed 30 times in an independent fashion.

3.3. Discussion of performance measures

Although many global performance measures can be found in the scientific literature, comparing EMOAs results is still an open problem. In contrast to single objective algorithms, the performance assessment of algorithms with multiple objectives constitutes a complex task. Among others, it involves quality of the outcome assessment (i.e., how to measure quality), computing resources used (e.g., time, number of function evaluations, etc.) as well as the analysis of several runs of the (stochastic-based) algorithm to take randomness and parameterization into account. Therefore, the analysis requires extensions of comparison methods. Instead of comparing objective vectors, approximation sets of several independent runs of algorithms need to be compared.

In multi-objective optimization, it is often impossible to know the (true) Pareto optimal set to be used in the comparison with the outcomes of EMOAs. Thus, general performance assessment criteria for multi-objective optimization algorithms should be considered including accuracy, coverage and variance, also called convergence, uniformity and spread. Under the best of circumstances, the obtained Pareto-optimal solutions are accurate, which means they are as close as possible to the true Pareto front of non-dominated solutions, well distributed and widely spread. Coverage and spread measures are closely related but are not exactly the same, because the former requires a representation of each region of the Pareto front, while the latter makes sure that the distance between points of the Pareto front approximation is evenly distributed (apparently it tends to give higher preference to boundary points).

In this context, theoretical and empirical techniques can be used for performance assessment. On the one hand, theoretical (analytical) approaches are usually difficult to adopt because of their limited scope (working effectively only with small data sets) and/or use of computational resources (many conditions should be checked). On the other hand, empirical (simulation) techniques are based in multiple runs of the algorithms under consideration, requiring the evaluation of standard parameters and the application of statistical testing procedures for comparing sets of Pareto front approximations.

In such a situation, and with the goal of accurately evaluating different approximation sets from multiple runs belonging to several stochastic multi-objective algorithms, complementary techniques can also be combined. To this end, we adopted both dominance-compliant quality indicators and 3D graphical representations of the reference fronts (composed of Pareto front solutions selected from all runs of an algorithm) for carrying out the performance assessment. While the former reduces each approximation set to a single quality value applying statistical tests to the samples, the latter shows the samples of the approximation sets giving information about how and where the performance differences occur.

Quality indicators allow the analysis of two algorithms to determine how much, and in which specific aspects, one of them is better than the other. However, these alternatives can only measure specific/limited quality aspects. Therefore, in our study we compute and analyze two complementary quality indicators that have reasonable properties related to the domain specific multi-objective optimization algorithms used in our experiments: SPREAD [36] and VUS [50,51].

The SPREAD indicator is commonly used for the comparison of different EMOAs. This indicator can be evaluated in either the objective or decision spaces, showing how far the Pareto front or set spreads in the objective or decision space, respectively. Hence, the larger the spread of the Pareto front is, the wider the range of values on objectives/variables it covers [52].

Volume Under the ROC hypersurface indicator (VUS) has a great significance in studying ML approaches for classification problems by the means of a ROC centered analysis. This indicator provides information on the volume under the convex hull, and can evaluate the solution set directly. Therefore, the better solution set is the one that obtains the higher value of VUS. The set of solutions on the ROC curve represents an approximation towards the set of optimal solutions (optimal ROC curve). However, although VUS closely resembles the commonly used hypervolume indicator, VUS is specific for the learning task. In particular, VUS considers the volume of the convex hull instead of the volume of the Pareto dominated subspace, working with a fixed reference point (also called the anti-ideal classifier). The reason for using VUS instead of the hypervolume indicator is that, for an ensemble of hard classifiers, it is always possible to create classifiers that have characteristics in the criterion space, which are given by the convex combinations of the objective function vectors of the classifiers in the ensemble. Therefore, VUS is the hypervolume indicator of the ensemble augmented by all these convex combinations.

Although the area under the (ROC) convex hull (AUC) has become a standard performance evaluation criterion in binary pattern recognition problems (being widely used to compare different classifiers independently of priors and costs), the AUC measure is only applicable to binary classification problems. The ROC curve (originally used in binary classification problems) was extended to a multi-class scenario in the work of Srinivasan and Srinivasan [53]. Moreover, some simpler generalizations to compute VUS [50] as well as more complex approaches to compute the ROC hypersurface (VUS) [54] were also proposed.

During several years, computational complexity and precision of MOEA trade-offs were discussed. In this regard, Edwards et al. [55] showed that the VUS value of a *near guessing* classifier is about the same as of a *near perfect* classifier when more than two classes are considered. Alternative definitions of VUS were subsequently introduced for dealing with such situations. In our work, we focus on finding sets of classifiers with optimal VUS, considering the simplified ROC proposed in the work of Landgrebe and Duin [51].

3.4. Results and discussion

This section introduces and discusses the results achieved from our experimental testbed, outlining relevant aspects concerning the behavior and performance of the proposed algorithms. The results analyzed in this section are organized following the two previously defined scenarios: parsimony maximization (3D-BinaryReal) and three-way classification (3D-Real).

In detail, in the first scenario we extend our previous study [21] by addressing not only the complexity of the classifier, but also the analysis of the rules as well as their type and relevance, with the goal of improving the classifier accuracy. Moreover, in the second scenario we test a three-way classification approach as an effective method to mark messages that need to undergo further examination by the user because of their low classification confidence.

On the one hand, optimization results achieved in the first scenario confirmed that the increment in the number of rules does not necessarily lead to an improvement in anti-spam filtering classification. On the other hand, increasing the number of anti-spam filtering rules has an impact on the classifier complexity, not only in the computational resources consumption for e-mail classification, but also in the administrators' ability to understand the filtering behavior and to maintain the anti-spam system.

From all the executed experiments, we can state that the classifier was able to reach high levels of accuracy in both FNR and FPR dimensions. In particular, it was found that FPR was close to zero even when using only 20% of the anti-spam rules. Additionally, the best accuracy trade-offs were also achieved using only those 20% of the available rules. From another complementary perspective, increasing the number of rules used in the classification process to exceed this 20% only produced a marginal impact on the classifier accuracy.

With the goal of better understanding those types of rules, we specifically studied the set of 20% anti-spam rules having the major impact in achieving the maximal classification accuracy. Table 4 presents the rank of the best rules (being used by all the algorithms in all the experiments), which are part of the best solutions of the reference Pareto front.

The rules shown in Table 4 are sorted according to their appearance in the reference Pareto front solutions (activation frequency). As we can see from Table 4, these rules are applied to both the e-mail header (e.g., messages origin domain) and body (i.e., message content). While the former is based on administrative measures and information sharing mechanisms between different anti-spam systems, the latter has a more customized nature according to language, economic area or activity of the institutions, and user pReferences

Moreover, the information shown in Table 4 indicates that 32.11% of relevant rules are related to the message body content and 6.5% are based on regular expressions, manually created by system administrators for parsing and checking e-mail structure, syntax and content. Remaining rules are related to e-mail message headers and different e-mail system administration policies.

As previously commented, for the second analyzed scenario (aiming at the minimization of FNR, FPR and UR) we provide a per-

Table 4
Rank of the 20% anti-spam rules being used in the best solutions (i.e., individuals) comprising the reference Pareto front.

Rules Name	Present in best individual solutions (%)	Present in best individual solutions (#)	body (message content) or header (meta-data) rules
BAYES_99	100.00%	99	body
SPF_HELO_FAIL	100.00%	99	header
T_LOTS_OF_MONEY	100.00%	99	header
BAYES_00	98.99%	98	body
FREEMAIL_FROM	96.97%	96	header
RDNS_NONE	96.97%	96	header
NO_DNS_FOR_FROM	92.93%	92	header
NORMAL_HTTP_TO_IP	83.84%	83	header
HTML_MESSAGE	82.83%	82	body
SPF_SOFTFAIL	81.82%	81	header
FROM_EXCESS_BASE64	80.81%	80	header
MISSING_MIMEOLE	78.79%	78	header
RCVD_HELO	71.72%	71	header
RCVD_NUMERIC_HELO	71.72%	71	header
BAYES_80	57.58%	57	body
RDNS_DYNAMIC	55.56%	55	header
RATWARE_MS_HASH	54.55%	54	header
BAYES_95	52.53%	52	body
IMPOTENCE	52.53%	52	body

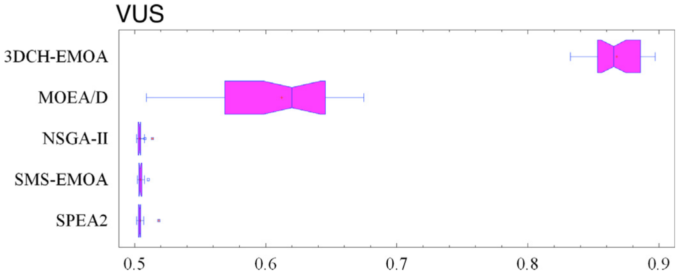


Fig. 8. VUS boxplot for the second analyzed scenario: minimization of FNR, FPR and unclassified ratio of e-mails in three-way classification.

formance assessment based on the graphical and indicator-based analysis of the Pareto front. In this way, boxplots depicting median, quartiles and outliers on the multi-criteria performance indicators (SPREAD and VUS) are shown for the five algorithms under consideration. The comparison of those algorithms is done with respect to the reference Pareto front, which is taken as a closest approximation of the true Pareto front.

In order to find statistically significant differences corresponding to VUS and SPREAD performance indicators among the MOEAs described above, and also taking into consideration that the underlying data do not fit a normal distribution, we performed a statistical analysis of the median differences by executing several Kruskal-Wallis tests.

In detail, Shapiro-Wilks tests were firstly carried out for the five EMOAs revealing low *p-values*, which allowed us to reject the hypothesis that the data come from a normal distribution, except for the 3DCH-EMOA approach with a p-value greater than 0.1 (for both indicators) and the NSGA-II alternative (for the SPREAD indicator). Tables 5 and 6 show the results of these Shapiro-Wilks tests corresponding to the VUS and SPREAD performance indicators, respectively.

After that, we used the Kruskal-Wallis test to check the null hypothesis that the medians within each of the five algorithms are the same. Since the p-value was less than 0.05 for both VUS and SPREAD indicators, we can confirm that there are statistically significant differences amongst the medians. In order to specifically show which medians are significantly different from each other,

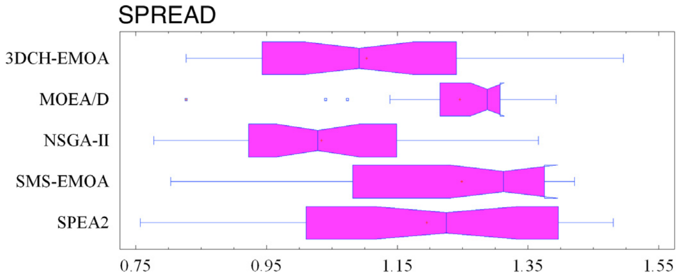


Fig. 9. SPREAD boxplot for the second analyzed scenario: minimization of FNR, FPR and unclassified ratio of e-mails in three-way classification.

Figs. 8 and 9 show the Box-and-Whisker plot corresponding to VUS and SPREAD indicators, respectively.

Complementarily, a comparison of the statistically significant differences amongst each pair of algorithms is also shown in Tables 7 and 8 for VUS and SPREAD indicators, respectively.

As expected, the three-objective CH-EMOA implementation (3DCH-EMOA) performs much better than all the other tested algorithms, presenting not only a high classification quality average, but also stable (predictable) behavior evidenced by a low VUS variance. This good performance is mainly motivated by the fact of CH-EMOA being an indicator-based algorithm, which precisely uses VUS as a selection criterion. The second position is occupied by the decomposition-based algorithm MOEA/D, showing a much worse VUS average and a higher variance, which is far from reaching the same performance level and stability obtained by the 3DCH-EMOA approach.

Additionally, the SPREAD indicator is useful for assessing the diversity of the solutions obtained by all the algorithms under consideration. As shown in Fig. 9, 3DCH-EMOA achieves a medium performance level with respect to this indicator. In fact, the average performance obtained by SMS-EMOA and MOEA/D algorithms is clearly better. The reason for this behavior is related to the fact that SMS-EMOA and MOEA/D can also approximate concave parts of the Pareto front. This is not allowed for the 3DCH-EMOA alternative and, as previously discussed, it is not necessary in the context of ROC performance. The relatively low variance of MOEA/D can be explained by the fact that it always uses the same set of reference points. A high variance in the results of both 3DCH-EMOA and SPEA2 approaches can be also observed. This last fact requires further investigation.

The 3D complementary plots of the reference Pareto front shown in Fig. 10 reveals the boundary between the dominated and non-dominated space (also known as attainment curve), where all the values are relative to the number of available rules (330) and the total number of messages (9349). Instead of representing the selected rates (i.e., FNR, FPR and UR) we used the absolute values to clearly show the practical impact of the decision alternatives.

With respect to Fig. 10, if the filter is forced to classify all the e-mails (using only 'legitimate' or 'spam' labels) the number of errors could be minimized to 13 misclassifications. Additionally, misclassifications could be further reduced up to 5 (in 20 out of 9349 cases) if the 'unclassified' label is allowed. Considering these results, the use of a three-way classification scheme seems very effective in reducing the number of classification errors.

Indeed, results shown in Fig. 11 confirm the effectiveness of maintaining a small number of e-mails unclassified, with the goal of increasing the classification quality. From Fig. 11 we can observe that the Pareto front with a few unclassified e-mails (20) dominates the Pareto front with less than 10 errors (3 FP and 3 FN). Therefore, we confirm the utility of those filters that keep messages unclassified when the computed solution has a low confidence level.

Table 5
Shapiro-Wilks tests for the VUS performance indicator.

	NSGA-II	SPEA2	SMS-EMOA	MOEA/D	3DCH-EMOA
Lowest *p*-value amongst the tests	0.0000168	1.88E-9	0.0354759	0.0148229	0.116923
Reject the hypothesis that the data comes from a normal distribution (confidence level)	99%	99%	95%	95%	lower than 90%

Table 6
Shapiro-Wilks tests for the SPREAD performance indicator.

	NSGA-II	SPEA2	SMS-EMOA	MOEA/D	3DCH-EMOA
Lowest *p*-value amongst the tests	0.593221	0.02838	0.000184025	0.0000983	0.268165
Reject the hypothesis that the data comes from a normal distribution (confidence level)	lower than 90%	95%	99%	99%	lower than 90%

Table 7
Kruskal-Wallis analysis corresponding to the VUS performance indicator.

	3DCH-EMOA	MOEA/D	NSGA-II	SMS-EMOA
SPEA2	DIF 95% conf. *p-value*: 2.872E-11	DIF 95% conf. *p-value*: 3.17E-11	NO DIF *p-value*: 0.813	NO DIF *p-value*: 0.214
SMS-EMOA	DIF 95% conf. *p-value*: 2.872E-11	DIF 95% conf. *p-value*: 3.175E-11	NO DIF *p-value*: 0.214	–
NSGA-II	DIF 95% conf. *p-value*: 2.872E-11	DIF 95% conf. *p-value*: 3.175E-11	–	–
MOEA/D	DIF 95% conf. *p-value*: 2.872E-11	–	–	–

Table 8
Kruskal-Wallis analysis corresponding to the SPREAD performance indicator.

	3DCH-EMOA	MOEA/D	NSGA-II	SMS-EMOA
SPEA2	NO DIF *p-value*: 0.0712747	NO DIF *p-value*: 0.756201	DIF 95% conf. *p-value*: 0.0034	NO DIF *p-value*: 0.604838
SMS-EMOA	DIF 95% conf. *p-value*: 0.000748948	NO DIF *p-value*: 0.169143	DIF 95% conf. *p-value*: 0.0000286438	–
NSGA-II	NO DIF *p-value*: 0.169143	DIF 95% conf. *p-value*: 0.00000133477	–	–
MOEA/D	DIF 95% conf. *p-value*: 0.000672439	–	–	–

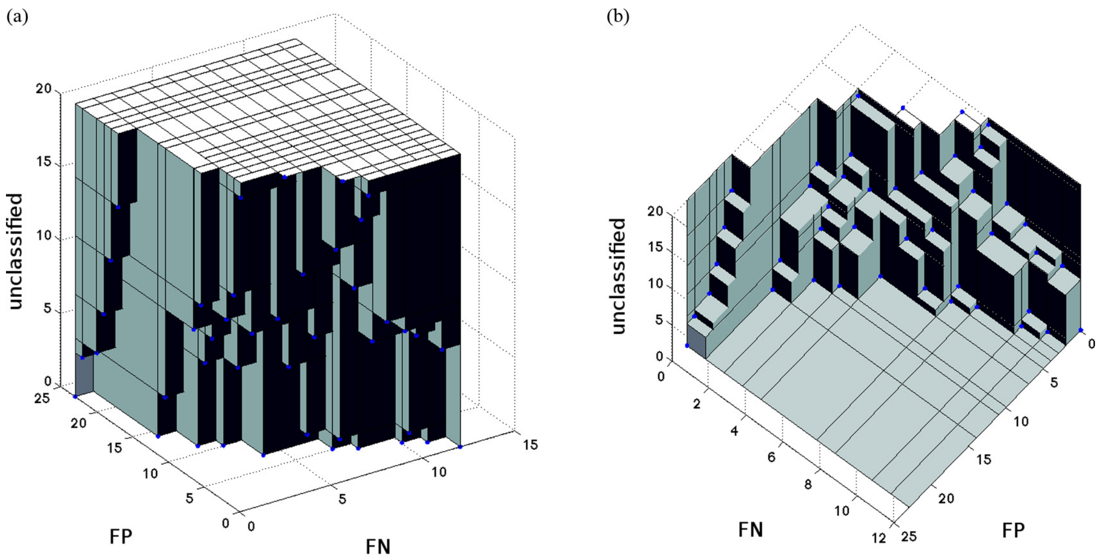


Fig. 10. 3D complementary plots of reference Pareto front for the second analyzed scenario.

Finally, to check the burden of these filter optimization methods, we measured both the overall time required to run the full experiments and also the relative burden of each EMOA. To this end, we executed the experiments in a quad-core Intel Xeon E5520 CPU at 2.27 GHz with 8GB RAM computer running Debian GNU Linux operating system. Table 9 shows the obtained results.

As shown in Table 9, NSGA-II, SPEA2 and MOEA/D execution times are very similar. These algorithms are approximately five times faster than the SMS-EMOA approach and thirty times faster than the 3DCH-EMOA alternative. The high computational burden of steady state methods such as SMS-EMOA is in this specific case worsened by the increased dimensionality (i.e., three objectives to minimize) of the optimization problem. The 3DCH-EMOA high

Table 9
Comparison of the execution times belonging to each algorithm.

	NSGA-II	SPEA2	SMS-EMOA	MOEA/D	3DCH-EMOA
Execution time per run (in seconds)	53.02	58.77	258.78	50.56	1759.70

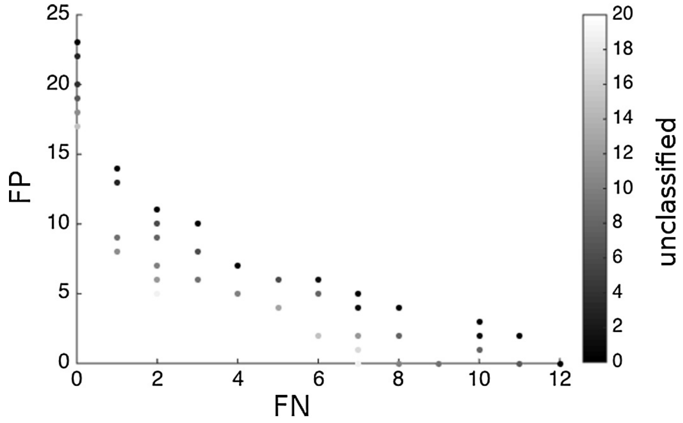


Fig. 11. Reference Pareto front for the second scenario showing the number of FN and FP errors as well as the number of unclassified messages.

computational burden is related to the convex hull calculation complexity, as described in Refs. [21,32]. MOEA/D is the algorithm that requires the smallest amount of computational time to compute an optimized SpamAssassin ruleset. However, the computational footprint of these EMOAs seems to limit their applicability in real domains. To cope with these issues, next subsection introduces a set of practical considerations to properly deploy these optimization techniques in real environments.

3.5. Practical deployment considerations

As previously discussed, the use of different MOEAs to optimize SpamAssassin filter scores presents an important computational footprint. As long as this issue should be taken into consideration to use them in real environments, we have compiled a set of recommendations to deploy score optimization mechanisms into production e-mail filtering servers.

First of all, the optimization process requires the use of target domain messages previously classified. As long as the filtering process in a Mail Transfer Agent (MTA) can be customizable through a configurable script, these e-mails can be easily compiled by modifying this script. Moreover, the classification could be also achieved by using SpamAssassin client (*spamc*). To cope with SpamAssassin misclassifications, two e-mail users (e.g., *not_spam* and *not_ham*) could be created to receive feedback from the final user. With this configuration, those messages compiled within an appropriate time period could be further used to execute the optimization process.

Complementarily, and keeping in mind the nature of the proposed methods, the optimization process should be periodically repeated once a month, or a week, depending on the specific computational capabilities. However, the proposed optimization process should not be implemented in a production e-mail filtering server to avoid lags in message exchanging.

Finally, with the goal of improving speed at affordable costs, the use of high performance parallel computing and cluster techniques (e.g., MapReduce, CUDA, etc.) is usually conducted. We found the application of these techniques suitable to achieve the improvement of MOEAs.

4. Conclusions and future work

In this work, we have evaluated the utility of several multi-objective evolutionary algorithms to optimize rule-based anti-spam filters from different but complementary perspectives. To this end, we presented two experimental case studies where filter complexity and three-way classification strategy were considered as additional objectives. The first scenario (parsimony maximization) revealed that the number of rules could be significantly reduced without affecting the filter performance. Moreover, experimental results related to the use of a three-way classification approach demonstrated the utility of defining a boundary region (where the classifier confidence is too low) to reduce the number of misclassification errors.

In this context, and from the experiments carried out, we would like to emphasize that from the 330 rules that match messages in the SpamAssassin corpus, only 5% to 20% of rules are really needed to achieve an optimal classification. Moreover, and taking into consideration the particular nature of the spam filtering domain, a considerable amount of relevant rules are based on regular expressions. These rules are used to specifically parse and check the e-mail structure, syntax and content, representing a major contribution in anti-spam filtering customization. The design of this type of rules constitutes an important share of the effort made by systems administrator to release novel and accurate anti-spam filters. Therefore, research aiming at the automatic generation of regular expressions from any given corpus is of high interest, having been initially addressed in the work of Basto-Fernandes et al. [56].

With regard to our three-way classification experiments, it was revealed that indicator-based algorithms perform well when carrying out multi-objective optimization of ROC curve performance. The best results for the VUS indicator were achieved by 3DCH-EMOA. Additionally, according to SPREAD indicator results, this algorithm also achieves good performance taking into account that this approach does not allow including points in the concave parts of the Pareto front. Finally, with the introduction of an extra ‘unclassified’ label in the filter (targeted to inform the user of those messages with a low confidence level), a considerable improvement in quality can be achieved to avoid harmful misclassifications at low cost for e-mail users (time).

Current and future work includes the investigation of whether obtained results generalize to data sets from other domains (e.g., web spam) where classification is commonly used. Moreover, as previously stated, the automatic generation of regular expression remains an interesting challenge in the domain of spam filtering.

All algorithms used in this study were implemented in JMetal Java framework and are available upon request by the authors.

Acknowledgments

This work has been partially funded by the [14VI05] Contract-Programme from the University of Vigo. SING group thanks CITI (*Centro de Investigación, Transferencia e Innovación*) from University of Vigo for hosting its IT infrastructure. The authors would like to thank Pu Wang for sharing the code of CH-MOGP.

References

[1] G.V. Cormack, Email spam filtering: a systematic review, Found. Trends Inf. Retr. 1 (4) (2007) 335–455.
[2] I. Androutsopoulos, J. Koutsias, K.V. Chandrinos, G. Paliouras, C.D. Spyropoulos, An evaluation of naive Bayesian anti-spam filtering, proceedings of the workshop on machine learning in the new information age, 11th European Conference on Machine Learning (2000) 9–17.
[3] D. DeBarr, H. Wechsler, Spam detection using random boost, Pattern Recognit. Lett. 33 (10) (2012) 1237–1244.
[4] H. Druker, S. Wu, V.N. Vapnik, Support vector machines for spam categorization, IEEE Trans. Neural Netw. 10 (5) (1999) 1048–1054.
[5] N. Pérez-Díaz, D. Ruano-Ordás, F. Fdez-Riverola, J.R. Méndez, SDAI: an integral evaluation methodology for content-based spam filtering models, Expert Syst. Appl. 39 (16) (2012) 12487–12500.
[6] J. Levine, Dns black lists and whitelists, in: Request For Comments 5782 (Informational), 2010 http://www.ietf.org/rfc/rfc5782.txt.
[7] The Spamhaus Project Ltd, The Spamhaus Project, 1998 http://www.spamhaus.org.
[8] Rhyolite Software LLC, Distributed Checksum Clearinghouses, 2000 http://www.rhyolite.com/dcc/.

[9] Apache Software Foundation, The Apache Spamassassin Project, 2003 http://spamassassin.apache.org/.
[10] I. Yevseyeva, V. Basto-Fernandes, D. Ruano-Ordás, J.R. Méndez, Optimising anti-spam filters with evolutionary algorithms, Expert Syst. Appl. 40 (10) (2013) 4010–4021.
[11] Rhyolite Software, Distributed Checksum Clearinghouses (2000) rhyolite.com/dcc/.
[12] V. Metsis, I. Androutsopoulos, G. Paliouras, Spam filtering with Naïve Bayes—which Naive Bayes? in: CEAS 2006—The Third Conference on Email and Anti-Spam, Mountain View, California, USA, 2006 www.ceas.cc/2006/15.pdf.
[13] I. Androutsopoulos, J. Koutsias, K.V. Chandrinos, G. Paliouras, C.D. Spyropoulos, An evaluation of naive bayesian anti-spam filtering, in: Proceedings of the workshop on machine learning in the new information age, 11th European Conference on Machine Learning, 2000, pp. 9–17.
[14] S. Kitterman, Sender policy framework (SPF) authentication failure reporting using the abuse reporting format, in: RFC 6652, 2012 https://tools.ietf.org/html/rfc6652.
[15] A. Back, Hashcash.org. Available at http://www.hashcash.org/ (1998).
[16] J. Fenton, Analysis of threats motivating domain keys identified mail (DKIM), in: Request For Comments 4686, 2006 http://www.ietf.org/rfc/rfc4686.txt.
[17] W.B. Cavnar, J.M. Trenkle, N-gram-based text categorization Proceedings of SDAIR-94, 3rd Annual Symposium on Document Analysis and Information Retrieval (1994) 161–175, http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.53.9367.
[18] J.R. Méndez, M. Reboiro-Jato, M. Díaz, E. Díaz, F. Fdez-Riverola, Grindstone4spam: an optimization toolkit for boosting e-mail classification, J. Syst. Software 85 (12) (2012) 2909–2920.
[19] I. Yevseyeva, V. Basto-Fernandes, J.R. Méndez, Survey on anti-spam single and multi-objective optimization, Proceedings of International Conference on ENTERprise Information Systems (2011) 120–129.
[20] V. Basto-Fernandes, I. Yevseyeva, J.R. Méndez, Optimization of anti-spam systems with multiobjective evolutionary algorithms, Int. Resour. Manage. J. 26 (1) (2012) 54–67.
[21] J. Zhao, V. Basto-Fernandes, L. Jiao, I. Yevseyeva, A. Maulana, R. Li, T.A. Bäck, M.T.M. Emmerich, Multiobjective optimization of classifiers by means of 3-d convex hull based evolutionary algorithm, ARXIV Comput. Sci. 1412 (2014) 5710, abs/1412.5710 http://arxiv.org/abs/1412.5710.
[22] J. Zhao, V. Basto Fernandes, L. Jiao, I. Yevseyeva, A. Maulana, R. Li, T. Back, K. Tang, M.T.M. Emmerich, Multiobjective optimization of classifiers by means of 3-D convex-hull-based evolutionary algorithms, Inf. Sci. 367–368 (2016) 80–104.
[23] Y. Yao, The superiority of three-way decisions in probabilistic rough set models, Inf. Sci. 181 (6) (2011) 1080–1096.
[24] B. Zhou, Y.Y. Yao, J.G. Luo, Cost-sensitive three-way email spam filtering, J. Intell. Inf. Syst. 42 (1) (2014) 19–45.
[25] B. Zhou, Y.Y. Yao, J.G. Luo, A three-way decision approach to email spam filtering, Proceedings of the 23th Canadian Conference on Artificial Intelligence (2010) 28–39.
[26] D. Beasley, Possible applications of evolutionary computation, in: Evolutionary Computation 1: Basic Algorithms and Operators, 1st edition, Institute of Physics Publishing, Bristol and Philadelphia, 2000, pp. 4–18 (Chapter 2).
[27] C.C. Coello, Evolutionary multi-objective optimization: a historical view of the field, IEEE Comput. Intell. Mag. 2 (2006) 28–36.
[28] K. Deb, Multi-Objective Optimization Using Evolutionary Algorithms, 1st edition, John Wiley & Sons, Chichester, UK, 2001.
[29] M. Laumanns, Analysis and applications of evolutionary multiobjective optimization algorithms, in: Ph.D. Thesis, Swiss Federal Institute of Technology, Zurich, 2003.
[30] A. Auger, J. Bader, D. Brockhoff, E. Zitzler, Hypervolume-based multiobjective optimization: theoretical foundations and practical implications, Theor. Comput. Sci. 425 (2012) 75–103.
[31] Y. Jin, Multi-objective machine learning Studies in Computational Intelligence, vol. 16, Springer-Verlag, 2006.
[32] P. Wang, M. Emmerich, R. Li, K. Tang, T. Bäck, X. Yao, Convex hull-based multi-objective genetic programming for maximizing receiver operating characteristic performance, IEEE Trans. Evol. Comput. 19 (2) (2014) 188–200.
[33] T. Fawcett, Rocgraphs: notes and practical considerations for researchers, in: Tech Report HPL-2003-4, HP Laboratories, 2004 http://www.purl.org/NET/tfawcett/papers/ROC101.pdf.
[34] N. Srinivas, K. Deb, Multiobjective optimization using nondominated sorting in genetic algorithms, Evol. Comput. 2 (3) (1994) 221–248.
[35] C. Fonseca, P. Fleming, Genetic algorithms for multi-objective optimization: formulation, discussion and generalization, Proceedings of the 5th International Conference on Genetic Algorithms (1993) 141–153.
[36] K. Deb, A. Pratap, S. Agarwal, T. Meyarivan, A fast and elitist multiobjective genetic algorithm: NSGA-II, IEEE Trans. Evol. Comput. 6 (2) (2002) 182–197.
[37] E. Zitzler, L. Thiele, Multiobjective optimization using evolutionary algorithms—a comparative case study, Proceedings of 5th Conference on Parallel Problem Solving from Nature (1998) 292–301.
[38] D.W. Corne, N.R. Jerram, J.D. Knowles, M.J. Oates, J. Martin, Pesa-II: region-based selection in evolutionary multiobjective optimization, Proc. Genet. Evol. Comput. Conf. (2001) 283–290.
[39] S. Mahfoud, Niching methods, in: Evolutionary Computation 2: Advanced Algorithms and Operators, 1st edition, Institute of Physics Publishing, Bristol and Philadelphia, 2000, pp. 87–93 (Chapter 13).
[40] O.M. Shir, M. Preuss, B. Naujoks, M. Emmerich, Enhancing decision space diversity in evolutionary multiobjective algorithms, Proceedings of the 5th International Conference on Evolutionary Multi-Criterion Optimization (2009) 95–109.
[41] E. Zitzler, S. Künzli, Indicator-based selection in multiobjective search, Proceedings of 8th International Conference on Parallel Problem Solving from Nature (2004) 832–842.
[42] M. Emmerich, N. Beume, B. Naujoks, An EMO algorithm using the hypervolume measure as selection criterion, Proceedings of the 3rd International Conference on Evolutionary Multi-Criterion Optimization (2005) 62–76.
[43] N. Beume, B. Naujoks, M. Emmerich, SMS-EMOA: multiobjective selection based on dominated hypervolume, Eur. J. Oper. Res. 181 (3) (2007) 1653–1669.
[44] D. Goldberg, Genetic Algorithms in Search, Optimization, and Machine Learning, Addison-Wesley, Reading, Massachusetts, 1989.
[45] J.J. Durillo, A.J. Nebro, F. Luna, E. Alba, On the effect of the steady-state selection scheme in multi-objective genetic algorithms, Proceedings of 5th International Conference on Evolutionary Multi-Criterion Optimization (2009) 183–197.
[46] Q. Zhang, H. Li, MOEA/D: a multi-objective evolutionary algorithm based on decomposition, IEEE Trans. Evol. Comput. 11 (6) (2007) 712–731.
[47] Apache Software Foundation, SpamAssassin Public Corpus, 2003 http://spamassassin.apache.org/publiccorpus.
[48] P. Resnick, Internet Message Format, 2008, RFC 5322 http://www.rfc-base.org/rfc-5322.html.
[49] J.J. Durillo, A.J. Nebro, jMetal: a java framework for multi-objective optimization, Adv. Eng. Software 42 (10) (2011) 760–771.
[50] C. Ferri, J. Hernández-Orallo, M.A. Salido, Volume under the roc surface for multi-class problems, Proceedings of the 14th European Conference on Machine Learning (2003) 108–120.
[51] T.C. Landgrebe, R.P. Duin, A simplified extension of the area under the roc to the multiclass domain, Proceedings of the 17th Annual Symposium of the Pattern Recognition Association of South Africa (2006).
[52] E. Zitzler, Evolutionary algorithms multiobjective optimization: methods and applications, in: Ph.D Thesis, 1999 http://www.tik.ee.ethz.ch/sop/publicationListFiles/zitz1999a.pdf.
[53] A. Srinivasan, A. Srinivasan, Note on the location of optimal classifiers in n-dimensional roc space, in: Technical Report PRG-TR-2-99, Oxford University Computing Laboratory, Oxford, 1999.
[54] T.C. Landgrebe, R.P. Duin, Efficient multiclass roc approximation by decomposition via confusion matrix perturbation analysis, IEEE Trans. Pattern Anal. Mach. Intell. 30 (5) (2008) 810–822.
[55] D.C. Edwards, C.E. Metz, R.M. Nishikawa, The hypervolume under the ROC hypersurface of 'near-guessing' and 'near-perfect' observers in n-class classification tasks, IEEE Trans. Med. Imaging 24 (3) (2005) 293–299.
[56] V. Basto-Fernandes, I. Yevseyeva, R.Z. Frantz, C. Grilo, N.P. Díaz, M. Emmerich, An automatic generation of textual pattern rules for digital content filters proposal, using grammatical evolution genetic programming, Procedia Technol. 16 (2014) 806–812.

Vitor Basto-Fernandes graduated in information systems in 1995, pos-graduated in distributed systems in 1997 and got his PhD on multimedia transport protocols in 2006 from the University of Minho—Portugal, where he has also been lecturer. From 2005 he has also been lecturing at the University of Tras-os-Montes e Alto Douro-Portugal, and invited assistant professor in the same university in 2007 and 2008. In 2008 he joined the Polytechnic Institute of Leiria-Portugal as assistant professor in the Informatics Engineering Department, where he is currently coordinator professor. His research interests include multi-objective optimization, semantic web and information security.

Iryna Yevseyeva is a Lecturer in Computer Science at the De Montfort University in Leicester, UK. Her expertise is in multicriteria optimization and decision analysis and their applications in various domains such as security, drug, discovery, manufacturing and health care. Previously, she was a post-doctoral researcher on security decision making at the Newcastle University, UK (2013–2016), on multi-objective optimization: for drug discovery at the Leiden University, Netherlands (2012–2013) and for spam filtering at the Polytechnic Institute of Leiria, Portugal (2011–2012); for scheduling at the INESC Porto, Portugal (2009–2011); for algorithms development at the University of Algarve, Portugal (2008–2009). Iryna received her PhD degree in computer science and optimization from the University of Jyvaskyla in Finland in 2007 for the research on multicriteria classification in healthcare.

José R. Méndez was born in Galicia (Spain) in 1977. Currently, he is an associate professor belonging to the computer science department of University of Vigo. He worked as a system administrator, software developer and IT (Information Technology) consultant in civil services and industry during 10 years. He is an active researcher belonging to SING group and his main interests include the development and improvement of anti-spam filters. (http://moncho.mdez-reboredo.info/)

Jiaqi Zhao received the B.Eng. degree in intelligence science and technology from Xidian University, Xi'an, China in 2010. He is currently pursuing the PhD degree in circuit and system from Xidian University, Xi'an China. Between 2013–2014, he was an exchange PhD student with the Leiden Institute for Advanced Computer Science (LIACS), University of Leiden, the Netherlands. He is currently a member of Key Laboratory of Intelligent Perception and Image Understanding of Ministry of Education, and International Research Center for Intelligent Perception and Computation, Xidian Universtiy, Xi'an, China. His current research interests include multi-objective optimization, machine learning and image processing.

Florentino Fdez-Riverola received a PhD in Computer Science from the University of Vigo in 2002. At present, he is Associate Professor at the University of Vigo and the Head of the SING research group (http://sing.ei.uvigo.es). Previously he was Sub-director of the Computer Science School (2003–06) and Director of the CITI (http://citi.uvigo.es) R&D Center (2009–10) at the University of Vigo. He has been a research collaborator with the BISITE group (http://bisite.usal.es) since 2002. He has led several Artificial Intelligence research projects sponsored by public and private institutions and has supervised seven PhD students. He is co-author of over 50 books, book chapters, journal papers, technical reports, etc. published by organisations such as Elsevier, IEEE, Ios Press, Kluwer, Springer Verlag, Morgan Kaufmann, etc.

Michael T. M. Emmerich, born 1973 in Coesfeld, Germany, received his Diploma in Applied Informatics and Chemical Engineering in 1999 from the University Dortmund, Germany. After working in industrial research companies, in 2003 he joined the Chair of Systems Analysis, Dortmund University. Here he received the doctorate degree in 2005 (promotor: H.P. Schwefel). Since 2005 he has been working as an Assistant Professor at the Leiden Institute for Advanced Computer Science, The Netherlands, where he leads the Multidisciplinary Optimization and Decision Analysis (MODA) research group. His main research interests are algorithm design for multicriteria optimization, spatial modeling, and complex systems analysis, with applications in engineering, chemistry, and the biomedical sciences.