Research Article

Boosting Accuracy of Classical Machine Learning Antispam Classifiers in Real Scenarios by Applying Rough Set Theory

N. Pérez-Díaz, D. Ruano-Ordás, F. Fdez-Riverola, and J. R. Méndez

*Higher Technical School of Computer Engineering, University of Vigo, Polytechnic Building,
Campus Universitario As Lagoas s/n, 32004 Ourense, Spain*

Correspondence should be addressed to F. Fdez-Riverola; riverola@uvigo.es

Received 11 March 2016; Revised 11 May 2016; Accepted 31 May 2016

Academic Editor: Fabrizio Riguzzi

Copyright © 2016 N. Pérez-Díaz et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Nowadays, spam deliveries represent a major problem to benefit from the wide range of Internet-based communication forms. Despite the existence of different well-known intelligent techniques for fighting spam, only some specific implementations of Naïve Bayes algorithm are finally used in real environments for performance reasons. As long as some of these algorithms suffer from a large number of false positive errors, in this work we propose a rough set postprocessing approach able to significantly improve their accuracy. In order to demonstrate the advantages of the proposed method, we carried out a straightforward study based on a publicly available standard corpus (SpamAssassin), which compares the performance of previously successful well-known antispam classifiers (i.e., Support Vector Machines, AdaBoost, Flexible Bayes, and Naïve Bayes) with and without the application of our developed technique. Results clearly evidence the suitability of our rough set postprocessing approach for increasing the accuracy of previous successful antispam classifiers when working in real scenarios.

1. Introduction and Motivation

Half a century ago, nobody could imagine the immense capabilities of current computing systems and network devices. Nowadays, they have drastically changed the way people share or exchange information and interact or communicate through a full Internet access (24 hours a day) implemented by last generation devices. Actually, most of the Internet consumers use the smartphone (67.5%) or tablet (42.3%) to access their e-mail accounts [1, 2].

As long as e-mail can be read everywhere at any time, spammers found this service particularly appropriate for delivering spam content. On the one hand, the usage of e-mail service has experienced an explosive growth achieving an average of 538.1 million messages sent daily during 2015, which represents an interannual increase of 5% since 2010 [3]. On the other hand, the percentage of spam e-mails suffered a slight reduction, representing an interannual decrease of 3.4% since 2010 [4]. Taking this situation into account, it is easy to realize that spam deliveries remain a problem to be solved in the modern society. To cope with this situation, the software industry (headed by Internet security enterprises)

has been continuously improving existing antispam filtering techniques and systems in order to enhance both filtering throughput [5–7] and classification accuracy.

Regarding classification accuracy, during the last decade, different research works have introduced the definition of several antispam domain authentication schemes (e.g., SPF [8] and RBL/RWL [9]), the description of novel collaborative approaches (e.g., DCC [10]), and the usage of diverse machine learning (ML) alternatives. In this connection, previous successful techniques such as Artificial Immune Systems (AIS) [11, 12], Case-Based Reasoning (CBR) systems [13, 14], different topologies of artificial neural networks (ANN) [15, 16], some simple but effective algorithms like k -NN [17, 18], Support Vector Machines (SVM) [19, 20], and different implementations of the well-known Naïve Bayes (NB) algorithm should be mentioned [21–23].

However, despite the large number of ML classifiers that have proven to be useful to fight against spam, only NB has been typically included by default in popular antispam filtering products such as SpamAssassin [24] and Wirebrush4SPAM [5], due essentially to its adequate balance

between the accuracy obtained and the associated computational cost [21, 22].

This is particularly true because in the antispam filtering domain the number of false positive (FP) errors made by the classifier while processing legitimate contents is of utmost importance [25]. This aspect still represents a major challenge for current techniques commonly applied in the area, especially when working in real and dynamic environments characterized by (i) the subjective nature of the spam concept, (ii) the adverse effects of concept drift, and (iii) the coexistence of multiple languages in individual mailboxes. To cope with this situation, Google (considered as one of the most valuable brands in the world [26]) decided to equip Gmail with a user-guided learning mechanism. As described in [27], this technology makes use of an ANN that takes into account the Gmail user classification criteria as feedback information for the neural network. In this context, it is obvious that the accuracy of this approach is directly proportional to the number of Gmail users. As a result, the large number of Gmail active accounts (more than 900 million in 2015 [28]) allows Gmail antispam filtering system to achieve a classification accuracy up to 99%. To this end, it is easy to realize that, due to its dependence on the number of users (to achieve suitable classification results), it can only be applied on e-mail services with a large number of active users.

As a direct consequence of the underlying operation mode, this strategy cannot be extrapolated to those e-mail services belonging to SMEs (*Small and Medium Enterprises*), since the number of e-mail users tends to be insufficient to achieve accurate classification rates. This situation has motivated SMEs to continue using typical antispam filtering frameworks such as SpamAssassin or Wirebrush4SPAM.

In such a situation, the continuous development and deployment of both exiting and novel antispam techniques over classical filtering frameworks continue to be a necessity for the SME environment. Specifically, we consider the reduction of type I (*false positive*) errors extremely important. To this end, in this work, we propose the use of rough sets (RS) theory due to its ability to deal with uncertainty and avoid type I errors [29].

In detail, RS theory was initially proposed by Pawlak in the 80s [30, 31], providing a formal methodology for the automatic transformation of data into knowledge [32]. The philosophy of this method is based in the supposition that any inexact concept (e.g., denoted by a class label) can be approximated superiorly and inferiorly using an indiscernibility relationship. As detailed in [33], one of the most important characteristics of RS theory is the ability to discover redundancy and dependencies between features.

Additionally, RS could provide interesting benefits to the correct classification of e-mails as they guarantee (i) effectiveness in discovering hidden patterns from data, (ii) the possibility of using both quantitative and qualitative information, (iii) capability to evaluate the significance of data, (iv) finding the minimal set of useful data that minimizes the overall classification complexity, (v) the automatic generation of a decision ruleset from scratch, and (vi) the identification of previously unknown relationships. All of these inherent features, together with some positive results achieved in previous

works [29], suggested to us the possibility of creating a RS postprocessing algorithm applicable to any ML classifier working as a standalone antispam filter. In this line, the present work introduces the proposal of a postprocessing algorithm and shows the viability of the idea from an experimental point of view.

While this section has introduced and motivated our proposal, the rest of the paper is organized as follows: Section 2 summarizes previous related approaches that also make use of RS theory in the antispam filtering domain. Section 3 details the developed algorithm that applies RS theory to extract domain specific decision rules from data, which will later guide the final revision of the initial proposed classification. Section 4 provides a clear description of the experimental protocol and documents the benchmark results obtained from the executed experiments. Finally, Section 5 provides conclusions and identifies future research work.

2. Related Work: Applying RS to Antispam Filtering

As previously stated, and mainly motivated by the massive proliferation of spamming activities, many researchers have studied the effectiveness of different approaches applied to the detection of illegitimate e-mails and other forms of spam [5, 8–25]. In this context, although several ML alternatives have been successfully used to categorize different e-mail corpora, recent studies have demonstrated the suitability of applying RS to specifically characterize messages comprising disjoint concepts (such as spam) [29].

In this line, Pérez-Díaz et al. [29] proposed three different execution schemes for using specific rules generated by applying RS theory. They compared these approaches against other well-known successful antispam techniques and reported a considerable reduction in the number of FP errors. Complementarily, Glymin and Ziarko [34] conducted a study to evaluate the use of variable precision RS (VPRS) [35] in the antispam filtering domain. In this work, a set of private Hotmail messages were collected during two years and VPRS were used to establish a decision table for classifying e-mails into two possible categories (i.e., spam or legitimate).

From a different perspective, some research studies focused their efforts on maintaining those rules generated through the use of RS [36–38]. These works proposed different frameworks to share generated rules from servers with the final goal of giving adequate support to a collaborative community interested in spam filtering. In the work of Chiu et al. [36], both the rule updating procedure and the policy for deleting obsolete rules are centralised in collaborative servers with the goal of immediately sharing available changes with the community. Additionally, the work of Lai et al. [37] introduces the generation of rules by means of RS, genetic algorithms, and reinforcement learning. Finally, the study carried out by Lai et al. [38] proposed novel methods to generate rules and validate their precision.

From another point of view, the work of Yang [39] proposed a framework (called RCFG) that combines RS and ant colony for applying an initial filtering to available data. Afterwards, the proposed approach uses a genetic algorithm

to carry out feature selection. Finally, different classifiers (i.e., SVM, k -NN, ANN, and NB) are used to identify spam e-mails.

Furthermore, there are also available several works that make use of RS to support three-way classification schemes. This type of alternative involves the definition of a third category (i.e., “suspicious”) to include those messages that cannot be easily classified as spam or legitimate. Following this approach, Zhao and Zhu [40] made use of the forward selection method [41] to generate a training corpus formed by eleven attributes and demonstrated the superiority of their VPRS-based algorithm when compared with Naïve Bayes. In the same line, the authors of [42, 43] initially reduced data attributes (also making use of the forward selection method), applying genetic algorithms for calculating RS reducts.

Complementarily, several researchers concentrated their efforts in applying the decision theoretic RS (DTRS) model to three-way classification [44, 45]. In DTRS, the two thresholds that differentiate spam (i.e., ham and suspicious) are initially calculated by using Bayesian theory in an automated way. Afterwards, classification with DTRS is made by means of a set of loss functions, which obtains the best classification with the minimal risk. In [44], a three-way decision model based on DTRS was compared with Naïve Bayes to evidence a reduction in error rates. Zhao et al. [45] proposed a novel approach based on α -positive-region of DTRS and compared achieved results with Naïve Bayes and other models based on RS.

Finally, Jia and colleagues [46, 47] enumerated the many benefits of three-way decision approaches and introduced a further challenge of discovering what to do with suspicious e-mails and how they can be examined in detail.

3. Using RS to Extract and Apply Domain Specific Decision Rules for Improving Accuracy

As can be seen from the last section, during last years a wide variety of contributions showing the applicability of RS [30–33] to the antispam filtering domain were presented. However, to the best of our knowledge, there is not a valid approach able to combine the fast execution speed of some successful ML classifiers with the good accuracy achieved by RS alternatives.

Therefore, in this work, we propose an innovative way to review the final output given by standard classifiers (in the form of a postprocessing algorithm) with the goal of reducing the number of type I (FP) errors. In this line, the generation of our complementary RS decision rules is carried out by using the same data (e-mail corpus) as in the case of the classifier (see Figure 1) but being applied only when a new incoming e-mail is initially classified as spam. By following this straightforward approach, our method becomes potentially applicable to any classifier.

As showed in Figure 1, the whole filtering process involves an initial feature extraction phase used to gather the specific values needed for representing a new incoming e-mail as an adequate input for the selected classifier. After that,

the classification model guesses the class of the message generating an initial output. In the case that the message was categorized as spam, it is further revised by our automatically generated RS decision rules before reaching a final classification. These revision rules are generated by our knowledge acquisition and representation module (showed in the right part of Figure 1), which is structured into two different stages: (i) feature selection and (ii) computation of RS rules.

In order to carry out the initial feature selection stage, a dense dataset should be generated from those messages that comprise the e-mail corpus. To do this, each column included in the dataset (condition attribute) a_i , $i \in (1 \cdots n)$ represents the existence or absence of a given token (i.e., the smallest portion of text enclosed by two characters included in $[[[:blank:]]]$ class) in the e-mail corpus. Therefore, the number of condition attributes of the newly generated dataset, n , is equal to the number of different tokens included in any message belonging to the e-mail corpus. Moreover, the real (known) class of each message (decision attribute) is also included as the last column of the dataset, being represented using a binary variable. In this context, the set of instances stored in the dataset is denominated universe, U , and its cardinality is equal to the number of messages finally represented, m .

During the feature selection stage, we perform a reduction of the dimensionality of the condition attributes that are part of the initial input dataset, represented by $A = \{a_1, \dots, a_n\}$. To this end, we apply two complementary procedures: (i) stop word removal and (ii) feature ranking. The first one comprises the elimination of those tokens having less than 3 characters and/or being included in the stop word list provided by Baeza-Yates and Ribeiro-Neto [48]. Then, we take advantage of Information Gain (IG) [49–51] to evaluate the suitability of each attribute included in the dataset. From all the available columns, we select the best 100 ranked attributes included in the dataset and discard the rest of the information [29]. Table 1 introduces an example of the result achieved after the execution of the feature selection stage, showing only 8 token attributes ($n = 8$) and 8 e-mails ($m = 8$) due to the lack of space. Additionally, we maintain the decision attributes (X) corresponding to the real (known) classes in the dataset (represented in the 9th column).

From the information stored in the dense dataset represented in Table 1, and applying RS theory, we designed a deterministic approach to generating a set of accurate revision rules [52], which will be later applied to the standard workflow represented in Figure 1. In this context, rule R establishes a specific combination of values for some condition attributes a_j (i.e., $R.conditions[a_2] = a_value_for_a_2 \wedge R.conditions[a_5] = a_value_for_a_5$) that determine a solution for a certain decision attribute x ($R.decision = solution$). The proposed algorithm able to carry out the rule extraction process is introduced in Algorithm 1. For representation purposes, a value of ‘?’ in a condition attribute, a_j , means that this feature should not be taken into consideration.

As showed in Algorithm 1, for each e-mail stored in the dataset, e_i , a new rule is generated through the computation of the shortest reduct (*computeShortestReduct* function) for a given concept ($X2$), which is defined as 1 for the same e-mail,

TABLE 1: Example of the reduced dense dataset (generated from the initial e-mail corpus) required for the computation of RS rules.

E	A								X
	a_1 Tax	a_2 Course	a_3 Student	a_4 Shipping	a_5 Price	a_6 Dear	a_7 Cialis	a_8 Levitra	x Class
e_1	0	0	0	1	1	0	1	0	1
e_2	0	1	1	0	0	0	0	0	0
e_3	1	1	1	0	0	1	0	0	0
e_4	0	0	0	1	0	0	0	1	1
e_5	0	0	0	0	0	0	1	1	1
e_6	1	1	0	1	0	1	0	0	0

FIGURE 1: Standard and augmented filtering process workflow executed whenever a new incoming e-mail arrives to the user mailbox.

'?' for messages of the same class, and 0 for other instances (lines (08)–(12) in Algorithm 1). In this context, a reduct is a minimal (irreducible) subset of features, $RED \subseteq A$, having the same precision to guess a concept (c) from the whole set of condition attributes in A . In order to assess the potential for classification of a set of condition attributes, $B \subseteq A$, all the instances, $\{e_1, \dots, e_m\}$, should be grouped into different subsets, where each subset contains all the indiscernible (indistinguishable) instances. In such a situation,

this grouping is known as the set of equivalence classes, $U/IND(B)$.

Two instances $e_j, e_k \in U$ are indiscernible regarding the condition attribute set, B , if they share the same values for all their attributes. Taking this into consideration, the potential for classification of the condition attributes included in B is measured by computing the lower approximation for the concept c , $\underline{B}c$. In this context, $\underline{B}c$ is the union of equivalence classes Y of $U/IND(B)$ having at least one positive instance

```

(00) FUNCTION computeRules E: MessageIdentifierVector,
(01)           A: ConditionAttributeMatrix, X: DecisionAttributeMatrix);
(02)   X2: DecisionAttributeMatrix;
(03)   RED: AttributeSet;
(04)   R: Rule;
(05)   RESULT: Ruleset;
(06)
(07)   FOREACH  $e_i$  INCLUDED IN E DO
(08)     FOREACH  $e_k$  INCLUDED IN E DO
(09)       IF ( $e_i == e_k$ ) THEN X2[ $e_k$ ] = 1;
(10)       ELSE IF ( X[ $e_i$ ] == X[ $e_k$ ] ) THEN X2[ $e_k$ ] = '?';
(11)       ELSE X2[ $e_k$ ] = 0;
(12)     END_FOREACH;
(13)   RED = computeShortestReduct (E, A, X2);
(14)
(15)   FOREACH  $a_j$  INCLUDED IN R DO
(16)     IF ( $a_j$  INCLUDED IN RED) THEN
(17)       R.conditions[ $a_j$ ] = A[ $e_i$ ,  $a_j$ ];
(18)     ELSE R.conditions[ $a_j$ ] = '?';
(19)   END_FOREACH;
(20)   R.decision=X[ $e_i$ ];
(21)   RESULT.add(R);
(22) END_FOREACH;
(23) RETURN RESULT;
(24) END_FUNCTION;

```

ALGORITHM 1: Pseudocode of the proposed algorithm for the generation of RS revision rules.

$c(e_j) = 1, e_j \in Y \in U/\text{IND}(B)$, and not any negative object $c(e_j) \neq 0, \forall e_i \in Y \in U/\text{IND}(B)$. Expression (1) shows the formal definition of the lower approximation of B for the decision concept c :

$$\underline{Bc} = \left\{ Y \in \frac{U}{\text{IND}(B)} : \forall e_i \in Y, c(e_i) \neq 0 \wedge \exists e_j \in Y, c(e_j) = 1 \right\}. \quad (1)$$

If we now consider the example shown in Table 1, as long as the fact that all the represented instances are discernible, $U/\text{IND}(A) = \{\{e_1\}\{e_2\}\{e_3\}\{e_4\}\{e_5\}\{e_6\}\}$, the lower approximation of concept x with attributes included in A is $\underline{Ax} = \{e_1, e_4, e_5\}$. Moreover, the subset of features $B = \{a_2\}$ is a reduct regarding concept x , because $U/\text{IND}(B) = \{\{e_1, e_4, e_5\}\{e_2, e_3, e_6\}\}$ and, hence, $\underline{Bx} = \{e_1, e_4, e_5\} = \underline{Ax}$.

Keeping in mind the existence of undefined values ('?') for concept $X2$ (considered in the algorithm shown in Algorithm 1), two lower approximations are equivalent if they only differ in those instances (e_i) having an undefined value for $X2$ concept.

Therefore, using the reference implementation of the proposed technique (refer to Additional-File1.java from the Supplementary Material available online at <http://dx.doi.org/10.1155/2016/5945192> for its Java implementation), we extracted the rules from the example data source included in Table 1. The extracted rules are shown as follows.

Revision Rules Generated by the Proposed Algorithm for the Example Shown in Table 1

- (2) IF $a_6 = \text{TRUE}$ THEN $x_1 = \text{FALSE}$
- (2) IF $a_7 = \text{TRUE}$ THEN $x_1 = \text{TRUE}$
- (2) IF $a_8 = \text{TRUE}$ THEN $x_1 = \text{TRUE}$
- (4) IF $a_8 = \text{FALSE}$ THEN $x_1 = \text{FALSE}$

As shown above, the rules generated by our proposed algorithm are simple and easy to execute. Therefore, the post-processing stage (labeled as RS-based decision in Figure 1) will not involve the usage of a great amount of computational resources. In addition, each rule generated by our algorithm includes the number of samples from training dataset that match with it (also known as coverage set cardinality). This information is very useful when a target message matches two or more conflicting rules. In this case, we use a voting scheme using the cardinality of the coverage set as vote weight. After that, if the obtained result is equal for both the spam and legitimate categories, the last one is selected for the target e-mail.

4. Model Benchmarking

In order to demonstrate the suitability of applying RS theory for improving the accuracy of previously successful ML classifiers in the antispam filtering domain, we designed an experimental protocol to execute our testbed. In Section 4.1, we include a description of this protocol introducing the reasons supporting our specific corpus selection, detailing

TABLE 2: Commonly used publicly available spam corpora.

Corpus	% legitimate	% spam	Number of messages
LingSpam ¹	83.3	16.6	2893
PU1 ¹	56.2	43.8	1099
PU2 ¹	80.0	20.0	721
PU3 ¹	51.0	49.0	4139
PUA ¹	50.0	50.0	1142
Spambase ²	39.4	60.6	4601
2005 TRECSspam ³	43.0	57.0	92189
2006 TRECSspam ³	35.0	65.0	37822
2007 TRECSspam ³	33.5	66.5	75419
SpamAssassin ⁴	74.5	25.5	9332

¹ Available at <https://labs-repos.iit.demokritos.gr/skel/i-config/downloads/>.

² Available at <http://ftp.ics.uci.edu/pub/machine-learning-databases/spambase/>.

³ Available at <http://trec.nist.gov/data/spam.html>.

⁴ Available at <https://spamassassin.apache.org/publiccorpus/>.

several preprocessing issues, and defining the fold cross validation scheme as well as different measures. Complementarily, in Section 4.2, we present and discuss the obtained results.

4.1. Experimental Protocol. With the goal of evidencing whether the combination of ML techniques with RS is adequate to reduce type I (FP) errors, we analyzed several publicly available datasets in order to select one able to ensure the validity of our experimental results. In this line, the most widespread are SpamAssassin [53], LingSpam [54], PU1 [54], PU2 [54], PU3 [54], PU4 [54], TREC [55–57], and Spambase from the UCI repository [58]. Table 2 compiles relevant information about these corpora including the percentage of legitimate and spam e-mails and the total number of available messages.

First of all, LingSpam corpus contains legitimate messages collected from a linguistic list merged with some spam messages directly compiled by its authors. It only includes 481 spam messages (16.6% of the total) and 2412 legitimate instances. Because of the small number of spam messages, most ML classifiers are affected by imbalanced learning [59] and, therefore, this dataset is not adequate for general experiments.

Secondly, PU1, PU2, PU3, and PUA corpora are distributed into 10 separate parts to facilitate the execution of 10-fold cross validation experiments [60]. As shown in Table 2, these corpora present different percentages of spam messages (43.8%, 20%, 49%, and 50%, resp.) making them appropriate to avoid the imbalanced data problem. However, due to the format used for their original representation, the usage of stop word lists, stemming, and other techniques based on gathering information from the e-mail header is not supported. As long as our approach requires the application of preprocessing techniques (e.g., usage of a stop word list), we have ruled out their use.

In the case of Spambase corpus, it contains 4601 messages (60.6% being spam) represented as feature vectors with information about 57 attributes. Due to the reduced

dimensionality (number of attributes) of this corpus, we found it unsuitable for the study.

Next, as described in Table 2, TREC conference presents three corpora grouped according to the mailing date (2005, 2006, and 2007, resp.) with different percentages of spam and ham messages (43%, 35%, and 33.5%, resp.). These corpora were built following the standard Internet message format (described in RFC-2822 [61]), keeping unaltered the original content of the messages. The preprocessing of the corpus does not include the detection and removal of duplicates.

Finally, SpamAssassin is one of the most used corpus by the antispam filtering community. It includes a total number of 9332 messages, of which 25.5% are spam e-mails. This standard corpus was built by the SpamAssassin developers without altering the original content of the messages. The preprocessing of this corpus (distributed in RFC-2822 format) included the removal of duplicates and the anonymization of specific data with the goal of guaranteeing receiver privacy. The ratio between the size of the corpus (medium-sized) and the proportion of spam and ham messages makes SpamAssassin corpus as the most suitable dataset for our experiments.

In order to demonstrate the benefits of our proposal in the antispam filtering domain, we selected four well-known and widely used ML classifiers: Naïve Bayes [62], Flexible Bayes [62], AdaBoost [63], and SVM [64–66]. Regarding their specific implementation, we chose the standard version of these classifiers included in the Weka Data Mining Software (available at <http://www.cs.waikato.ac.nz/~ml/weka/>). To successfully use Naïve and Flexible Bayes Weka implementations, the dimensionality of the input feature vectors was limited to 1000 characteristics (using IG feature ranker). Moreover, Naïve Bayes classifier was executed using binary features (0/1) while Flexible Bayes was evaluated with continuous attributes (frequency). Additionally, AdaBoost was configured to use Decision Stumps as metaclassifiers and 150 boosting iterations. Complementarily, using IG method, we reduced the dimensionality of input vectors down to 700 binary features. Finally, a 1-degree polynomial function was selected as kernel for SMO algorithm (Weka SVM implementation), which was executed using binary feature vectors with a size of 2000 (reduced using IG feature ranker).

All these parameters were established taking into consideration the integral evaluation methodology proposed by Pérez-Díaz et al. [25] for accurately ranking different content-based spam filtering models. Additionally, in the work of Méndez et al. [49], IG showed the best performance for all the compared models, while in [25] the authors experimentally computed the best number of features (using the IG feature ranker) for all the available classifiers. Finally, with the goal of ensuring the validity of our results, all the experiments were conducted under a stratified 10-fold cross validation schema [60].

To correctly assess the performance achieved by applying our RS revision method when compared to the independent execution of ML classifiers, we have chosen four groups of well-known measures: (i) percentage of correctly classified messages, false positive and false negative (FN) errors, (ii) f -score (also known as F_1 score or f -measure) [67, 68], (iii) balanced f -score [68], and (iv) Total Cost Ratio (TCR) [22].

TABLE 3: Performance gain obtained by the use of the proposed RS-based approach when compared to the initial output of standard ML classifiers.

	% OK	% FP	% FN
NB (+RS)	90.80 (+3.91)	5.08 (−4.83)	4.13 (+0.92)
FB (+RS)	88.86 (−0.19)	0.26 (−0.21)	10.89 (+0.41)
AB (+RS)	94.03 (+0.84)	1.66 (−1.48)	4.31 (+0.64)
SVM (+RS)	94.48 (+0.01)	0.88 (−0.79)	4.64 (+0.78)

TABLE 4: F -score and balanced f -score rates for different β values.

	$\beta = 0.25$	$\beta = 0.5$	$\beta = 1$
NB (+RS)	0.81 (+0.165)	0.81 (+0.130)	0.82 (+0.063)
FB (+RS)	0.94 (+0.010)	0.86 (+0.001)	0.72 (−0.009)
AB (+RS)	0.92 (+0.057)	0.91 (+0.041)	0.88 (+0.012)
SVM (+RS)	0.95 (+0.031)	0.93 (+0.018)	0.88 (−0.004)

4.2. Obtained Results and Discussion. By applying the experimental protocol defined in the previous section, we straightforwardly evaluate the suitability of our proposed approach to improve the performance of different widely recognized ML classifiers. In this context, Table 3 shows the percentage analysis of the different type of errors (FP and FN) as well as the hits achieved by the analyzed ML techniques, giving specific information about the performance gain obtained by the use of the proposed RS-based approach. As described in Section 3, RS rules are automatically applied to revise the output of each ML classifier when it initially classifies a given message as spam.

As initially shown in Table 3, the percentage of correct classifications (% OK) using ML techniques was improved when RS revision rules were applied with the only exception of Flexible Bayes algorithm. The particular behavior of Flexible Bayes classifier can be explained by the very high number of FN errors, which cannot be successfully addressed by our proposal that is only applied in those cases in which an incoming e-mail is initially classified as spam. In the light of these results, the overall combination of ML techniques with the proposed revision approach was able to reduce the number of misclassifications of legitimate e-mails. This behavior avoids the incorrect filtering of relevant messages for the end user with a minimal footprint in FN errors (ability to detect spam).

With the goal of having a more insightful perspective about these initial results, we also computed f -score and balanced f -score values, merging recall and precision for different β alternatives. Table 4 presents the obtained results.

As shown in Table 4, the combination of precision and recall measures with the same weight ($\beta = 1$) evidences slightly worse results when applying RS in combination with Flexible Bayes and SVM. However, this assumption is unrealistic from a real user perspective for which classification errors own a very different importance. In this line, Table 4 reveals that when increasing the penalization of type I (FP) errors (using lower values of β), the RS-based revision approach achieves great evaluation results.

In this context, and with the goal of providing a further analysis about the real impact of type I errors from a cost-sensitive point of view, we carried out TCR evaluations for all the analyzed models. These results are shown in Figure 2.

As clearly shown in Figure 2(a), if the cost of an FP error is considered as important as a FN misclassification ($\lambda = 1$), SVM and Flexible Bayes classifiers do not achieve additional benefits. However, a significant improvement is obtained by the application of our automatic revision procedure when working in real scenarios (situation modeled by assigning to λ different values).

5. Conclusions and Future Work

In this work, we have presented a RS-based postprocessing technique able to reduce type I (FP) errors made by different well-known classifiers previously applied in the antispam filtering domain. To this end, we have designed a straightforward algorithm able to extract simple and complementary revision rules exploiting the same corpus used to train the original classifiers. Our approach is only applied to those messages initially classified as spam, alleviating the use of valuable computational resources in real implementations.

Results achieved by the execution of the experimental protocol have demonstrated the effectiveness of our proposal for improving the performance of different ML classifiers. Particularly, different cost-sensitive measures (such as TCR or balanced f -score) obtained accurate rates for our RS-based revision approach when dealing with type I errors. The main advantage of its combined execution is an increase on classification hits, which is an important issue to augment the final classifier user experience.

Moreover, the impact on the time required for carrying out the final classification when our proposed method is applied is negligible because (i) the postprocessing is not applied on each classification (only for messages initially classified as spam) and (ii) the time and computer resources needed to evaluate the matching of rules are very low. Additionally, the knowledge acquisition and representation process represented in Figure 1 (as well as the training of the standard ML classifiers) can be executed in a different machine with the goal of saving computational resources on the hardware used to deploy the antispam filter.

The main drawback of our approach is the deterministic nature of the generated revision rules. In this regard, Pawlak and colleagues [52] have shown the limitations of RS deterministic approaches when compared to probabilistic ones that work with information uncertainty inherent in many classification problems (such as spam). Additionally, the main advantage of probabilistic models lies on providing a unified approach for both deterministic and nondeterministic knowledge representation systems. Taking this idea into account, our main line of future research work includes searching for complementary probabilistic approaches able to generate rules that outperform the capabilities of our current algorithm. Moreover, in order to complement our current work, we also find interesting the identification of novel feature selection and extraction methods. To this end, we believe that regular expressions representing more than one token

FIGURE 2: TCR evaluation varying the importance assigned to type I errors for the analyzed models.

could be more effective than features made up of a single one. Finally, we also found interesting the idea of carrying out the dynamic validation of rules in order to detect when they became obsolete.

Competing Interests

The authors declare that there are no competing interests regarding the publication of this paper.

Acknowledgments

This work has been partially funded by (i) the 14VI05 Contract-Programme from the University of Vigo, (ii) the INOU15-06 Project from the University of Vigo, and (iii) *Agrupamento INBIOMED from DXPCTSUG-FEDER unha maneira de facer Europa* (2012/273). SING group thanks CITI (*Centro de Investigación, Transferencia e Innovación*) from University of Vigo for hosting its IT infrastructure.

References

- [1] J. van Rijn, "The ultimate mobile email statistics overview," 2015, <http://www.emailmonday.com/mobile-email-usage-statistics>.
- [2] J. Jordan, *53% of Emails Opened on Mobile, Email Testing and Email Marketing Analytics—Litmus*, 2015, <https://litmus.com/blog/53-of-emails-opened-on-mobile-outlook-opens-decrease-33>.
- [3] The Radicati Group Inc, *A Technology Market Research Firm, Email Statistics Report, 2013–2017*, 2015, <http://www.radicati.com/wp-content/uploads/2013/04/Email-Statistics-Report-2013-2017-Executive-Summary.pdf>.
- [4] Statista, *Global Email Spam Rate 2012–2015*, 2016, <http://www.statista.com/statistics/270899/global-e-mail-spam-rate/>.
- [5] N. Pérez-Díaz, D. Ruano-Ordas, F. Fdez-Riverola, and J. R. Méndez, "Wirebrush4SPAM: a novel framework for improving efficiency on spam filtering services," *Software—Practice and Experience*, vol. 43, no. 11, pp. 1299–1318, 2013.
- [6] D. Ruano-Ordás, J. Fdez-Glez, F. Fdez-Riverola, and J. R. Méndez, "Effective scheduling strategies for boosting performance on rule-based spam filtering frameworks," *Journal of Systems and Software*, vol. 86, no. 12, pp. 3151–3161, 2013.
- [7] D. Ruano-Ordás, J. Fdez-Glez, F. Fdez-Riverola, and J. R. Méndez, "Using new scheduling heuristics based on resource consumption information for increasing throughput on rule-based spam filtering systems," *Software—Practice and Experience*, 2015.
- [8] S. Göring, "An overview of the Sender Policy Framework (SPF) as an anti-phishing mechanism," *Internet Research*, vol. 17, no. 2, pp. 169–179, 2007.
- [9] J. M. M. da Cruz, *Spam: Classement Statistique de Messages Électroniques: Une Approche Pragmatique*, Presses des Mines, 2012.
- [10] Rhyolite Inc, *Distributed Checksum Clearinghouses*, 2015, <http://www.rhyolite.com/dcc/>.
- [11] J. Timmis, A. Hone, T. Stibor, and E. Clark, "Theoretical advances in artificial immune systems," *Theoretical Computer Science*, vol. 403, no. 1, pp. 11–32, 2008.
- [12] J. Timmis, T. Knight, L. N. de Castro, and E. Hart, "An overview of artificial immune systems," in *Computation in Cells and Tissues*, pp. 51–91, Springer, Berlin, Germany, 2004.
- [13] S. J. Delany, P. Cunningham, A. Tsybmal, and L. Coyle, "A case-based technique for tracking concept drift in spam filtering," *Knowledge-Based Systems*, vol. 18, no. 4–5, pp. 187–195, 2005.
- [14] F. Fdez-Riverola, E. L. Iglesias, F. Díaz, J. R. Méndez, and J. M. Corchado, "SpamHunting: an instance-based reasoning system for spam labelling and filtering," *Decision Support Systems*, vol. 43, no. 3, pp. 722–736, 2007.
- [15] C.-H. Wu, "Behavior-based spam detection using a hybrid method of rule-based techniques and neural networks," *Expert Systems with Applications*, vol. 36, no. 3, part 1, pp. 4321–4330, 2009.
- [16] A. H. Mohammad and R. A. Abu Zitar, "Application of genetic optimized artificial immune system and neural networks in spam detection," *Applied Soft Computing Journal*, vol. 11, no. 4, pp. 3827–3845, 2011.
- [17] S. Jiang, G. Pang, M. Wu, and L. Kuang, "An improved K-nearest-neighbor algorithm for text categorization," *Expert Systems with Applications*, vol. 39, no. 1, pp. 1503–1509, 2012.
- [18] X. Zhou, Y. Hu, and L. Guo, "Text Categorization based on Clustering Feature Selection," *Procedia Computer Science*, vol. 31, pp. 398–405, 2014.
- [19] V. Mitra, C.-J. Wang, and S. Banerjee, "Text classification: a least square support vector machine approach," *Applied Soft Computing Journal*, vol. 7, no. 3, pp. 908–914, 2007.
- [20] H. Drucker, D. Wu, and V. N. Vapnik, "Support vector machines for spam categorization," *IEEE Transactions on Neural Networks*, vol. 10, no. 5, pp. 1048–1054, 1999.

- [21] V. Metsis, I. Androustopoulos, and G. Paliouras, "Spam filtering with Naïve bayes—which Naïve bayes?" in *Proceedings of the 3rd Conference on Email and Anti-Spam (CEAS '06)*, July 2006.
- [22] I. Androustopoulos, J. Koustias, K. V. Chandrinou, G. Paliouras, and C. Spyropoulos, "An evaluation of naïve Bayesian anti-spam filtering," in *Proceedings of the 11th European Conference on Machine Learning. Workshop on Machine Learning in the New Information Age*, pp. 9–17, Barcelona, Spain, 2000.
- [23] M. Sahami, S. Dumais, D. Heckerman, and E. Horvitz, "A Bayesian approach to filtering junk e-mail," Tech. Rep. WS-98-05, AAI Press, 1998.
- [24] SpamAssassin Group, *The Apache SpamAssassin Project*, 2015, <http://spamassassin.apache.org/>.
- [25] N. Pérez-Díaz, D. Ruano-Ordás, F. Fdez-Riverola, and J. R. Méndez, "SDAI: an integral evaluation methodology for content-based spam filtering models," *Expert Systems with Applications*, vol. 39, no. 16, pp. 12487–12500, 2012.
- [26] Forbes, *The World Most Valuable Brands*, 2015, <http://www.forbes.com/powerful-brands/list/>.
- [27] Official Gmail Blog, "The mail you want, not the spam you don't," 2015, <https://gmail.googleblog.com/2015/07/the-mail-you-want-not-spam-you-dont.html>.
- [28] F. Lardinois, *Gmail Has Now 900M Active Users*, 2015, <http://techcrunch.com/2015/05/28/gmail-now-has-900m-active-users-75-on-mobile/>.
- [29] N. Pérez-Díaz, D. Ruano-Ordás, J. R. Méndez, J. F. Gálvez, and F. Fdez-Riverola, "Rough sets for spam filtering: selecting appropriate decision rules for boundary e-mail classification," *Applied Soft Computing Journal*, vol. 12, no. 11, pp. 3671–3682, 2012.
- [30] Z. I. Pawlak, *Rough Sets: Theoretical Aspects of Reasoning About Data*, Kluwer Academic, New York, NY, USA, 1991.
- [31] Z. I. Pawlak, J. Grzymala-Busse, R. Slowinski, and W. Ziarko, "Rough sets," *Communications of the ACM*, vol. 38, no. 11, pp. 88–95, 1995.
- [32] Z. I. Pawlak, "Rough sets," *International Journal of Computer & Information Sciences*, vol. 11, no. 5, pp. 341–356, 1982.
- [33] Z. Pawlak, "Rough sets: present state and the future," *Foundations of Computing and Decision Sciences*, vol. 18, no. 3-4, pp. 157–166, 1993.
- [34] M. Glymin and W. Ziarko, "Rough set approach to spam filter learning," *Proceedings of the International Conference of Rough Sets and Intelligent System Paradigms (RSEISP '07)*, vol. 4585, pp. 350–359, 2007.
- [35] W. Ziarko, "Variable precision rough set model," *Journal of Computer and System Sciences*, vol. 46, no. 1, pp. 39–59, 1993.
- [36] Y.-F. Chiu, C.-M. Chen, B. Jeng, and H.-C. Lin, "An alliance-based anti-spam approach," in *Proceedings of 3rd International Conference of Natural Computation (ICNC '07)*, pp. 203–207, August 2007.
- [37] G.-H. Lai, C.-M. Chen, C.-S. Lai, and T. Chen, "A collaborative anti-spam system," *Expert Systems with Applications*, vol. 36, no. 3, pp. 6645–6653, 2009.
- [38] G. Lai, C. Chou, C. Chen, and Y. Ou, "Anti-spam filter based on data mining and statistical test," *Computer and Information Science*, vol. 208, pp. 179–192, 2009.
- [39] Y. Yang, "A novel framework based on rough set, ant colony optimization and genetic algorithm for spam filtering," *International Journal of Advancements in Computing Technology*, vol. 4, no. 14, pp. 516–525, 2012.
- [40] W. Zhao and Y. Zhu, "Classifying email using variable precision rough set approach," in *Rough Sets and Knowledge Technology*, G.-Y. Wang, J. F. Peters, A. Skowron, and Y. Yao, Eds., vol. 4062 of *Lecture Notes in Computer Science*, pp. 766–771, Springer, 2006.
- [41] D. C. Whitley, M. G. Ford, and D. J. Livingstone, "Unsupervised forward selection: a method for eliminating redundant variables," *Journal of Chemical Information and Computer Sciences*, vol. 40, no. 5, pp. 1160–1168, 2000.
- [42] W. Zhao and Z. Zhang, "An email classification model based on rough set theory," in *Proceedings of the International Conference on Active Media Technology (AMT '05)*, pp. 403–408, May 2005.
- [43] W. Zhao and Y. Zhu, "An email classification scheme based on decision-theoretic rough set theory and analysis of email security," in *Proceedings of the IEEE Region 10 Conference (TENCON '05)*, pp. 1–6, Melbourne, Australia, November 2005.
- [44] B. Zhoy, Y. Yao, and J. Luo, "A three-way decision approach to email spam filtering," in *Advances in Artificial Intelligence: 23rd Canadian Conference on Artificial Intelligence, Canadian AI 2010, Ottawa, Canada, May 31–June 2, 2010. Proceedings*, vol. 6085 of *Lecture Notes in Computer Science*, pp. 28–39, Springer, Berlin, Germany, 2010.
- [45] C. Zhao, W. Zeng, M. Jiang, and Z. He, "A decision-theoretic rough set approach to spam filtering," in *Proceedings of the 10th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD '13)*, pp. 130–134, July 2013.
- [46] X. Jia and L. Shang, "Three-way decisions versus two-way decisions on filtering spam email," in *Transactions on Rough Sets XVIII*, J. F. Peters, A. Skowron, T. Li, Y. Yang, J. Yao, and H. S. Nguyen, Eds., vol. 8449 of *Lecture Notes in Computer Science*, pp. 69–91, Springer, 2014.
- [47] X. Jia, K. Zeng, W. Li, T. Liu, and L. Shang, "Three-way decisions solution to filter spam email: an empirical study," in *Rough Sets and Current Trends in Computing: 8th International Conference, RSCTC 2012, Chengdu, China, August 17–20, 2012. Proceedings*, vol. 7413 of *Lecture Notes in Computer Science*, pp. 287–296, Springer, Berlin, Germany, 2012.
- [48] R. Baeza-Yates and B. Ribeiro-Neto, *Modern Information Retrieval*, Addison Wesley, 1999.
- [49] J. R. Méndez, F. Fdez-Riverola, F. Díaz, E. L. Iglesias, and J. M. Corchado, "A comparative performance study of feature selection methods for the anti-spam filtering domain," in *Advances in Data Mining. Applications in Medicine, Web Mining, Marketing, Image and Signal Mining: 6th Industrial Conference on Data Mining, ICDM 2006, Leipzig, Germany, July 14–15, 2006. Proceedings*, vol. 4065 of *Lecture Notes in Computer Science*, pp. 106–120, Springer, Berlin, Germany, 2006.
- [50] J. R. Méndez, I. Cid, D. Glez-Peña, M. Rocha, and F. Fdez-Riverola, "A comparative impact study of attribute selection techniques on naïve bayes spam filters," in *Advances in Data Mining. Medical Applications, E-Commerce, Marketing, and Theoretical Aspects: 8th Industrial Conference, ICDM 2008 Leipzig, Germany, July 16–18, 2008 Proceedings*, vol. 5077 of *Lecture Notes in Computer Science*, pp. 213–227, Springer, Berlin, Germany, 2008.
- [51] J. R. Méndez, E. L. Iglesias, F. Fdez-Riverola, F. Díaz, and J. M. Corchado, "Analyzing the impact of corpus preprocessing on anti-spam filtering software," *Research on Computing Science*, vol. 17, pp. 129–138, 2005.
- [52] Z. Pawlak, S. K. M. Wong, and W. Ziarko, "Rough sets: probabilistic versus deterministic approach," *International Journal of Man-Machine Studies*, vol. 29, no. 1, pp. 81–95, 1988.
- [53] SpamAssassin, *SpamAssassin Public Corpus*, 2003, <https://spamassassin.apache.org/publiccorpus/>.

- [54] I. Androustopoulos, G. Paliouras, and E. Michelakis, “Learning to filter unsolicited commercial e-mail,” Tech. Rep. 2004/2, NCSR “Demokritos”, 2004.
- [55] G. Cormack and T. Lynam, “TREC 2005 spam track overview,” in *Proceedings of the 14th Text REtrieval Conference (TREC '05)*, November 2005.
- [56] G. Cormack, “TREC, 2006 spam track overview,” in *Proceedings of the 15th Text REtrieval Conference (TREC '06)*, pp. 117–127, November 2006.
- [57] G. V. Cormack, “TREC 2007 spam track overview,” in *Proceedings of the 16th Text REtrieval Conference (TREC '07)*, Gaithersburg, Md, USA, November 2007.
- [58] S. Hettich, C. L. Blake, and C. J. Merz, “UCI Repository of machine learning databases,” 1998, <http://archive.ics.uci.edu/ml/datasets/Spambase>.
- [59] H. He and E. A. Garcia, “Learning from imbalanced data,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 21, no. 9, pp. 1263–1284, 2009.
- [60] R. Kohavi, “A study of cross-validation and bootstrap for accuracy estimation and model selection,” in *Proceedings of the 14th International Joint Conference on Artificial Intelligence*, pp. 1137–1143, 2004.
- [61] P. Resnick, *RFC2822—Internet Message Format*, 2001, <https://www.ietf.org/rfc/rfc2822.txt>.
- [62] G. H. John and P. Langley, “Estimating continuous distributions in bayesian classifiers,” in *Proceedings of the 11th Conference on Uncertainty in Artificial Intelligence (UAI '95)*, pp. 338–345, 1995.
- [63] Y. Freund and R. E. Schapire, “Experiments with a new boosting algorithm,” in *Proceedings of the 13th International Conference on Machine Learning (ICML '96)*, pp. 148–156, 1996.
- [64] V. N. Vapnik, *The Nature of Statistical Learning Theory*, Springer, New York, NY, USA, 1995.
- [65] J. Platt, “Fast training of support vector machines using sequential minimal optimization,” in *Advances in Kernel Methods—Support Vector Learning*, B. Schoelkopf, C. Burges, and A. Smola, Eds., pp. 41–65, The MIT Press, 1998.
- [66] S. S. Keerthi, S. K. Shevade, C. Bhattacharyya, and K. R. K. Murthy, “Improvements to Platt’s SMO algorithm for SVM classifier design,” *Neural Computation*, vol. 13, no. 3, pp. 637–649, 2001.
- [67] D. M. W. Powers, “Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation,” *International Journal of Machine Learning Technology*, vol. 2, no. 1, pp. 37–63, 2011.
- [68] C. J. V. Rijsbergen, *Information Retrieval*, Butterworth-Heinemann, 1979.