Personalized Spam Filtering with Natural Language Attributes

Rushdi Shams and Robert E. Mercer
Department of Computer Science
University of Western Ontario
London ON N6A 5B7, Canada
Email: rshams@csd.uwo.ca, mercer@csd.uwo.ca

Abstract—Email spam is one of the biggest threats to today’s Internet. To deal with this threat, many anti-spam filters have been developed. One big challenge for these filters is to predict the labels of emails in a personalized mailbox. In this paper, we report the performance of an anti-spam filter named SENTINEL. In addition to some commonplace attributes, SENTINEL uses attributes related to natural language stymetly. The filter has been tested with six benchmark datasets in the Enron-Spam collection. Classifiers generated by well-known meta-learning algorithms like ADABOOSTM and BAGGING perform equally the best, while a Random Forest (RF) generated classifier performs almost as well. The performance of classifiers using Support Vector Machine (SVM) and Naïve Bayes (NB) are not satisfactory. Comparisons show that the performance of SENTINEL surpasses that of a number of state-of-the-art personalized filters proposed in previous studies.

I. INTRODUCTION

Today’s spam emails are very different than what they were called in the 1980s. Simply, they are not just “Usenet messages cross-posted to numerous newsgroups” anymore [1]. Now, spams are in fact used to refer to unsolicited, massively posted commercial and non-commercial emails. The onslaught of spams has grown exponentially. In March 2013, for instance, 100 billion spams were sent out daily. This quantity is 98% more than that from the previous quarter [2]. The effects of spams include loss of individual productivity, financial loss of organizations, cluttering of user inboxes, and consumption of network bandwidth. Therefore, spam filtering is an important challenge.

Spam filtering is an interesting binary classification problem. Classifiers are trained on a reasonable quantity of spams and hams (i.e., legitimate emails), and then applied on unseen emails to classify them into one of these two categories. Classifiers induced by well-known algorithms like Naïve Bayes (NB) [3][4][5], Random Forest (RF) [6], Support Vector Machine (SVM) [7] and Neural Networks (NN) [8] are used in many operational anti-spam filters. As test beds for these classifiers, several email datasets have been used most of which contain emails that are collected from random sources. Enron-Spam [3], on the other hand, is a collection of six benchmark datasets that is useful to test personalized filters, i.e., filters that are trained on incoming messages of a particular user that they are intended to protect.

Natural language attributes of email subject and body have a considerable ability to discern spams and hams [3][8]. Most of these attributes are based on the importance of a term in the email (i.e., term frequency or TF) and its rarity in the email dataset (i.e., inverse document frequency or IDF). The TF-IDF attributes are very promising for personalized filters (see, for example, [3] and [9]), whereas Lai and Tsai [5] report that TF-IDF does not perform as well on randomly collected emails as on personalized emails. Furthermore, the calculation of TF-IDF is done on the word-count space. Therefore, for each newly arrived email, this attribute needs to be re-calculated. The re-calculation, if not done incrementally, can introduce latency. As well, the use of terms by spammers changes over time [10]. Therefore, the value of attributes based solely on terms and their frequency may decrease because of this time-sensitivity. However, there are natural language attributes that are still unexplored in this problem domain, viz. readability, grammar and spelling mistakes, and the use of function and content words. Similar attributes have been proposed by writer stymetly and used to detect fraud in documents [11].

In this paper, we report the performance of an anti-spam filter named SENTINEL on personalized emails. Besides conventional spam filtering attributes, SENTINEL uses many natural language attributes that are related to email readability, grammar, spelling, use of function and content words, TF-IDF, etc. The attributes are extracted from six datasets of the Enron-Spam collection. In this experiment, we test the classifiers generated by SENTINEL using Random Forest (RF), Support Vector Machine (SVM), and Naïve Bayes (NB). In addition, two more classifiers are generated using the meta-algorithms ADABOOSTM and Bootstrap Aggregating (BAGGING). We use RF as the base classifier for both and name these classifiers BOOSTED RF and BAGGED RF, respectively. Results show that BOOSTED RF and BAGGED RF perform almost equally the best while the performance of RF is a close second. Interestingly, the performance of SVM depends on the quantity of spams in the training set. NB has the poorest results of all—which is understandable as most of the attributes exhibit inter-dependency. Comparisons show that the performance of SENTINEL surpasses that of a number of state-of-the-art personalized filters proposed in previous studies. The natural language attributes of SENTINEL are similar to stymetlogic attributes and are therefore language independent. As a result, the filter may be an excellent means to classify spam emails in any language.

The next section details the related work. Following that, we describe the materials and methods in Section III. In Section IV, we report the experimental findings. Finally, Section V concludes the paper.
II. RELATED WORK

The earliest of the NB anti-spam filters were simple and computationally efficient. Likewise, they attained low misclassification rates on several datasets. These filters exploited the simple Bayesian framework, a set of rules, and both header and content attributes [12]. The initial success of these filters led several others to emerge by simply replacing the rules with predictive models (see, for example, [3][5][13][14]). Several variations of these filters include multivariate Bernoulli, Bag of Words (BoW), multinomial boolean, etc. Metsis et al. [3], for instance, used 3000 multinomial boolean TF attributes—the results were very impressive. Two years later, they achieved even better results by using the transformed TF attributes [15]. Nowadays, the performance of an NB filter is considered as the de facto standard to compare a newly developed method.

SV filters are efficient for training in much the same way as NB filters nevertheless they need incremental training to reduce latency [16]. Furthermore, for this problem domain, SV filters can handle large attribute sets [7]. Most of the benchmark SVM filters use the frequency-based linear kernel. As well, there are several SVM filters that use updatable supervised clustering algorithms [17]. However, the trade-off with SVM filters is that their misclassification rate for personalized emails is quite high.

The advantages of using meta-learning anti-spam filters are manifold. Firstly, when a base learner with a sufficient tree depth is used, they achieved low misclassification rates on many public datasets [18]. Secondly, these filters are resistant to the problem of overfitting and therefore they gain a more appropriate accuracy even on an unbalanced dataset [19]. However, these filters have the weakness of ensemble learning—the interpretation of results is difficult. Studies show that meta-learning filters outperform many decision tree, NB and SVM filters. Surprisingly, the use of this category of filters is still not as widespread as NB and SVM filters.

Over the last decade, Artificial Immune System based anti-spam filters have become a popular choice. These filters use detectors on the email for pattern matching. Detectors are in fact regular expressions that are defined a priori. Each detector is given a weight that is adjusted as the filters recognize a pattern in a given email. The weights of the matching detectors are then used (usually combined) to determine the email’s class label. Notable immune system based filters are reported in [9], [20], and [21]. As the filters seek specific signatures in the emails, they are widely used in personalized email classification where similar patterns can be found in the writing style of the person the filter intends to protect. Also, many of these filters are able to deal with Concept Drift—the gradual or abrupt change of thematic context over time such as new advertisement themes in spam.

III. MATERIALS AND METHODS

A. Email Datasets

For decades, several email datasets have been used to gauge the performance of anti-spam filters, viz. LingSpam, SpamAssassin, and CSDMC2010. Because the emails in these datasets are collected randomly and therefore do not represent personal inboxes, anti-spam filters aimed at protecting a particular user cannot be evaluated on them. Simply put, to evaluate personalized spam filtering, we need personalized email data. Enron-Spam2 [3] is a collection of emails composed of six datasets each containing ham emails from a single person in the Enron corpus. These ham collections are dubbed as follows: farmer-d, kaminski-v, kitchen-l, williams-w3, beck-s, and lokay-m. The spams are collected from three different sources. First, a mix of spams that are collected from the SpamAssassin corpus and spam traps of the Honeypot project are put together; these spams are dubbed as (SH). Second, BG spams are collected from the spam traps of Bruce Guenter. Third, spams collected randomly from the mailbox of Georgios Paliouras (GP). The foregoing six ham email collections are each paired with one of the three spam collections (SH, BG, and GP). Thereafter, the six collections are dubbed as Enron 1 to Enron 6. Of the six collections, Enron 1–3 are ham skewed (ham:spam is 3:1) while Enron 4–6 are spam skewed (ham:spam is 1:3). The summary of the characteristics of the Enron-Spam dataset can be found in Table I.

For conservative estimates of the proposed filter, the following pre-processing steps are considered. First, the SUBJECT field of spams can contain symbols such as $ or !. As well, they contain spam words such as porn, webcamp, or lottery. Therefore, such entries are excluded from the SUBJECT fields of the emails. Second, many emails in the collection can contain an ATTACHMENT field. This extraneous field is removed from the emails (if any). Third, non-ASCII characters in the email text are removed since the values of our natural language attributes are affected by their presence.

B. Attribute Selection

Each email in our experiment is represented as $(\vec{x}, y)$, where $\vec{x} \in \mathbb{R}^n$ is a vector of $n$ attributes and $y \in \{\text{spam}, \text{ham}\}$ is the label of the email. In our study, we explored 37 attributes to classify spam emails and therefore, $n = 37$. Table II summarizes the attributes used in our experiment.

1) Word-level Attributes: To calculate the word-level attributes, we treated each email as a bag of words. We curated a dictionary that comprises 381 spam words. Using this dictionary, we counted the frequency of spam words in the emails. This attribute is inspired by the interesting findings of Graham [13] who showed that merely finding the word

<table>
<thead>
<tr>
<th>Enron + Spam</th>
<th>Ham:Spam</th>
<th>Time-stamp (Ham, Spam Duration)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enron 1(kitchen-l + BG)</td>
<td>300/2500</td>
<td>(1239, 0.35)</td>
</tr>
<tr>
<td>Enron 2(kaminski-v + SH)</td>
<td>300/1400</td>
<td>(11299, 0.30)</td>
</tr>
<tr>
<td>Enron 3(kitchen-l + BG)</td>
<td>4912/1200</td>
<td>(201, 2502)</td>
</tr>
<tr>
<td>Enron 4(williams-w3 + GP)</td>
<td>1500/4500</td>
<td>(301, 2502)</td>
</tr>
<tr>
<td>Enron 5(beck-s + SH)</td>
<td>1500/700</td>
<td>(1000, 300)</td>
</tr>
<tr>
<td>Enron 6(lokiay-m + BG)</td>
<td>1500/4500</td>
<td>(600, 300)</td>
</tr>
</tbody>
</table>

TABLE I: The details of the Enron-Spam collection as described by Metsis et al. [3]: composition, ham-spam ratio, and time-stamps of the emails.

\footnote{Downloadable at \url{https://labs-repos.iit.demokritos.gr/skel/i-config/downloads/enron-spam}}

\footnote{Consult with \url{http://www.projecthoneypot.org}}

\footnote{Overview at \url{http://untroubled.org/spam}}

\footnote{Downloadable at \url{http://cogenglab.csd.uwo.ca/sentinel/spam-term-list.html}}

\footnote{Most of the public email datasets are unbalanced [16].}
click in the emails can detect 79.7% of spam emails in a dataset with only 1.2% ham misclassification rate. Our other word-level attributes include the frequency of alpha-numeric words, verbs, and function words. To identify the verbs in the emails, we used the Stanford part-of-speech Tagger\(^6\). On the other hand, to identify function words, we utilized the stoplist function of a topic indexer named Maui\(^7\).

In our experiment, we included the term frequency (TF) attribute used by at least two NB filters [3][15]. For each term, which appears in at least four training emails, information gain scores are computed according to previous work [22]. Of all the terms in the dataset, 3000 with the highest scores are considered as the TF vector for all the emails. Thereafter, should any of these 3000 terms be found in an email, the term’s corresponding frequency value in the TF vector is first transformed (see [15] for details) and then normalized. Finally, the normalized value is used to calculate the probability score of the email according to Bayes’ theorem. The overall method can be found in the study of Kosmopoulos et al. [15].

Furthermore, we measured the TF-ISF of each email. Here, the term frequency (TF) is the square root of the frequency of a term \(t\) in an email \(M\) while the inverse sentence frequency (ISF) is a relative measure of whether \(t\) is common or rare in \(M\). TF-ISF, therefore, reflects how important each \(t\) is to \(M\). In addition, the measure controls the fact that in an email, some terms are generally more common than others. In contrast, we used TF-IDF of each email as our attribute, which is a numerical statistic that reflects how important a term \(t\) is to an email \(M\) in the dataset \(D\) to which \(M\) belongs. The TF-IDF value increases proportionally to the number of times \(t\) appears in the email \(M\), but is offset by the frequency of \(t\) in the dataset \(D\). The definition of TF is the same as for TF-ISF while inverse document frequency (IDF) measures whether a given term \(t\) is common or rare in the dataset \(D\).

2) Error Attributes: Our next set of attributes is related to the grammar and spelling errors present in the emails. For each email, we simply counted the frequency of grammatical and spelling errors using a Java API called LanguageTool\(^6\). By summing up the values of these two attributes, we introduced a third attribute in this category named Language Errors. Interestingly, the error attributes are not normalized since we have obtained better results without normalizing them.

3) Readability Attributes: Readability deals with the difficulty of reading a sentence, a paragraph or a document. At the heart of readability lies the notion of simple and complex words. Simple words are those that have at most two syllables while complex words contain three or more syllables. Since both types of words have a significant contribution for text readability, we have included them in our attribute list. Five standard scores use these two types of words to determine the readability of a given text: Fog index, Smog index, Flesch reading ease score, Forcast, and Flesch-Kincaid index. For each email, all five readability scores are computed and used as attributes. Among the scores, Fog index measures the relative use of complex words in a document. In addition, we modified the Fog index formula to measure the relative use of simple words in a document, and considered that as an attribute, too. Furthermore, we considered the arithmetic inverse of Fog index as another attribute. The other attributes in this category include email length (i.e., total sentences in an email), average word length (i.e., total syllables over total terms in an email), and TF-IDF of simple and complex words. The details of the attributes are discussed elsewhere [23].

Note that the aforementioned attributes are computed by both including and excluding the function words in the emails—the exceptions being the frequency of function words, TF, email length, and TF-IDF attributes. Analyzing the real valued attributes, we have found that their distribution is either left-tailed or right-tailed. This skewness of attribute values, which leads to poor classifiers, has been eliminated by using a logarithmic transformation of all the attribute values. Once log-transformed, the distribution of the attributes becomes normal. For each attribute, the transformed values are then normalized by the maximum; the resulting normalized values are therefore in \([0, 1]\).

C. Learning Algorithms

In this experiment, well-known learning algorithms such as Random Forest (RF), Naïve Bayes (NB), and Support Vector Machine (SVM) are used to induce binary classifiers. We also used two meta-learning algorithms viz. ADABOOSTM1 and BAGGING. Random Forest has been chosen as the weak learner for each, and are named BOOSTED RF and BAGGED RF, respectively. The reasons for choosing these algorithms follow.

As found in much recent research [4][6], RF is able to produce highly accurate spam classifiers, which is desirable for any anti-spam filter. Overall, RF classifiers also have the reputation for being fast and efficient with large data [24]. On the other hand, although it generates complex models, SVM is a popular choice for anti-spam filters. The algorithm has notable performances with header features [4][5][6][7]. NB is also a widely used learning algorithm for anti-spam filters. It is simple yet provides powerful spam detectors [3][6]. In addition, on many occasions, NB using simple attributes such as TF-IDF has even outperformed quality learning algorithms, like SVM. Unlike the others, ADABOOSTM1 and BAGGING are meta-learners that improve a given weak learning algorithm most of the time [25][26]. In the following experiments, we have considered RF as the weak learner for these two algorithms. Both ADABOOSTM1 and BAGGING are simple, fast, and above all, are less susceptible to overfitting the training.

<table>
<thead>
<tr>
<th>Category</th>
<th>Quantity</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Word-level Attributes</td>
<td>11</td>
<td>Spam words, Alpha-numeric words, Function words, Verbs, TF [3][15], TF-IDF, TF-IDF.</td>
</tr>
<tr>
<td>Error Attributes</td>
<td>3</td>
<td>Grammar and spelling mistakes.</td>
</tr>
<tr>
<td>Readability Attributes</td>
<td>23</td>
<td>Simple and complex words and their TF-IDF, Fog index, Simple and Inverse Fog index, Smog index, Flesch reading ease score, Forcast, Flesch-Kincaid score, Email length, Word length.</td>
</tr>
</tbody>
</table>
TABLE III: Confusion matrix for spam classification problem.

<table>
<thead>
<tr>
<th>Prediction</th>
<th>Actual</th>
<th>Spam</th>
<th>Ham</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spam</td>
<td>(n_{s\rightarrow s})</td>
<td>(n_{s\rightarrow h})</td>
<td></td>
</tr>
<tr>
<td>Ham</td>
<td>(n_{h\rightarrow s})</td>
<td>(n_{h\rightarrow h})</td>
<td></td>
</tr>
</tbody>
</table>

D. Evaluation Measures

All of the measures reported in this paper depend on the confusion matrix given in Table III. Precision is the fraction of spam predictions that are correct and can be written as follows:

\[ \text{PREC} = \frac{n_{s\rightarrow s}}{n_{s\rightarrow s} + n_{h\rightarrow s}}. \]

On the other hand, recall—also known as spam recall—examines the fraction of spam emails being retrieved:

\[ \text{REC} = \frac{n_{s\rightarrow s}}{n_{s\rightarrow s} + n_{s\rightarrow h}}. \]

F-score (FM), simply, is the harmonic mean of precision and recall. Accuracy, on the other hand, is the percentage of correctly identified spams and hams:

\[ \text{ACC} = \frac{n_{s\rightarrow s} + n_{h\rightarrow s}}{n_{s\rightarrow s} + n_{s\rightarrow h} + n_{h\rightarrow s} + n_{h\rightarrow h}}. \]

Noting that users might accept some spams to enter into their inbox but that they prefer their hams not end up in the spam-traps, email misclassification is cost-sensitive. The ham misclassification (false positive) rate denotes the fraction of all ham emails classified as spams:

\[ \text{FPR} = \frac{n_{h\rightarrow s}}{n_{h\rightarrow s} + n_{s\rightarrow h}}. \]

In contrast, the spam misclassification (false negative) rate is the fraction of all spams delivered to the user inbox:

\[ \text{FNR} = \frac{n_{s\rightarrow h}}{n_{s\rightarrow h} + n_{s\rightarrow s}}. \]

Another viable alternative for the cost-sensitive analysis of email misclassification is to report the Area Under Curve (hereinafter, AUC), measured using the Receiver Operating Characteristics (ROC) curves. The ROC curve is a 2D graph whose Y-axis represents 1 – FNR and whose X-axis represents FPR, thereby depicting the compromises between the cost of \(n_{s\rightarrow h}\) and \(n_{h\rightarrow s}\).

E. Experimental Procedure

Treating each dataset independently, from each email of the six Enron-Spam datasets, SENTINEL extracts the real-valued attributes using its text processing unit. Using a conventional stratified 10-fold cross-validation approach, the filter then generates for each dataset five classifiers using the five algorithms described in Section III-C. The classifiers are then evaluated. In a K-fold cross-validation, the original dataset is treated as a single fold. Each dataset is then divided into \(K\) equal-sized folds or subsets. Then each classifier is trained on \(K-1\) folds and evaluated on the remaining fold. Stratification means that the class (i.e., ham or spam) in each fold is represented in approximately the same proportion as in the full dataset. The cross-validation process is then repeated until each of the \(K\) folds is used exactly once as the validation data. The final estimation of the classifier is the average of the \(K\) results from the folds.

IV. RESULTS AND DISCUSSIONS

To evaluate anti-spam filters on Enron-Spam, scores described in Section III-D are almost always averaged across the six datasets [3]. The Arithmetic Mean is the standard averaging method used, regardless of the skewness present in the datasets (see Section III-A), but we have found that due to the presence of extreme data points, the Harmonic Mean provides a more appropriate estimate than the arithmetic mean. That said, in this section, we report the harmonic mean of the results found from the six datasets.

Measures such as precision, recall, and AUC of the classifiers on Enron-Spam are presented in Table IV. The notable result from the data is that RF, BOOSTED RF, and BAGGED RF perform evenly good, which is confirmed by a student t-test with \(\alpha = 0.05\). The tests further reveal that the first three classifiers in Table IV are better than SVM and NB at \(\alpha = 0.05\). The best scores achieved by SENTINEL are highlighted in the chart—except for recall, BAGGED RF has the best numbers. Interestingly, the NB classifier has a better AUC compared to that of SVM at \(\alpha = 0.05\); for the remaining cases, however, the latter performs the better. The near-perfect AUCs of the first three classifiers strongly suggest that SENTINEL did not achieve the results by merely guessing everything either as spams (i.e., high \(n_{s\rightarrow s}\) but low \(n_{s\rightarrow h}\)) or hams (i.e., high \(n_{s\rightarrow h}\) but low \(n_{h\rightarrow s}\))—the filter was properly balanced during its labeling. The accuracy and F-score of the classifiers can be found in Figures 1a and 1b, where they are compared to four state-of-the-art personalized filters: LC [9], NB-BOW [27], SVM-BOW [27], and ICRM [20]. Again, the first three classifiers of SENTINEL perform about equally well and surpass the rest. The filter that can claim to be a reasonably close second is the LC filter inspired by artificial immune systems [9]. For these two measures, the differences between SENTINEL’s optimal filter (BAGGED RF), and LC are significant at \(\alpha = 0.05\).

The average FPR and FNR of SENTINEL can be found in Figures 1c and 1d, respectively. It comes as no surprise that BAGGED RF and BOOSTED RF perform the best—BAGGED RF misclassifies hams the least (FPR=2.1\%), see Figure 1c) while BOOSTED RF misclassifies spams the least (FNR=0.7\%, see Figure 1d). In addition, SENTINEL is compared to three personalized filters that are considered to be yardsticks: NB-TF [3], BAYESIAN [14], and NB-SLWE [28]. Except for the NB-TF filter, the three best classifiers of SENTINEL significantly outperform these yardsticks. Four of SENTINEL’s classifiers (the exception is NB) surpass the FNR of NB-TF, but only BAGGED RF outperforms the FPR of NB-TF while our second best classifier—BOOSTED RF—closely ties with NB-TF; the FPR of BAGGED RF is lower than NB-TF but the difference is significant only at \(\alpha = 0.10\).

We further investigated the FPR and FNR achieved by our...
classifiers on each of the six datasets (see Figure 2). Interestingly, RF exhibits strength similar to the meta-learners for spam classification (Figure 2b) but its ability to identify hams diminishes when more spams are included in its training data (Figure 2a)—specifically, for Enron 5 and Enron 6, even NB outperforms it. SVM’s performance shows that this classifier’s training data should be carefully selected—with spam skewed training data its ham misclassification rate is as high as 17%. This experiment also illustrates that the skewness of data has an effect on the training of anti-spam filters—except for the aberrant trend displayed by the NB classifier likely due to the presence of dependency among the attributes [23], the remaining classifiers misclassify fewer hams in Enron 1–3 (ham skewed) and fewer spams in Enron 4–6 (spam skewed).

V. CONCLUSIONS

In this paper we describe the development and evaluation of a prototype personalized anti-spam filter named SENTINEL. The filter uses natural language attributes, the majority being connected to stylistic aspects of writing. We evaluate the filter with a benchmark personalized email collection called Enron-Spam. Five well-known learning algorithms that induce binary classifiers using the real-valued natural language attributes of the emails are explored. The experimental outcomes show that SENTINEL performs best with two meta-learners: BOOSTED RF and BAGGED RF. As well, the performance of SENTINEL surpasses that of a number of state-of-the-art personalized filters proposed in previous studies. The evaluation therefore indicates that the filter can be utilized as a personalized anti-spam filter.

We still need to investigate several aspects of the filter viz. its real-time training and response latency. Moreover, its performance on Concept Drift can be observed by substituting the spams of Enron-Spam collection with the latest data. These investigations are left as future work.

ACKNOWLEDGMENTS

Support for this work was given by Natural Sciences and Engineering Research Council of Canada (NSERC) Discovery Grant to Robert E. Mercer. We are indebted to Vangelis Metsis and Aris Kosmopoulos for their correspondences regarding the use of their TF attribute.

REFERENCES

Fig. 2: Ham and Spam misclassification rates of the classifiers on the Enron-Spam collection.

(a) Ham misclassification rates (FNR) of the classifiers for the six datasets.

(b) Spam misclassification rates (FPR) of the classifiers for the six datasets.