An Improved Bayesian with Application to Anti-Spam Email

ZHAN Chuan, LU Xian-liang, ZHOU Xu, HOU Meng-shu
(School of Computer Science and Engineering, UESTC Chengdu 610054 China)

Abstract Along with the wide application of e-mail nowadays, many spam e-mails flood into people's email-boxes and cause catastrophes to their study and life. In anti-spam e-mails campaign, we depend on not only legal measures but also technological approaches. The Bayesian classifier provides a simple and effective approach to discriminate classification. This paper presents a new improved Bayesian-based anti-spam e-mail filter. We adopt a way of attribute selection based on word entropy, use vector weights which are represented by word frequency, and deduce its corresponding formula. It is proved that our filter improves total performances apparently in our experiment.

Key words word entropy; Bayesian classification; anti-spam e-mail filter; attribute selection; vector

Spam e-mail is annoying to most users, as they waste users time, money, network bandwidth as well as clutter users’ mailboxes. It can even be harmful, e.g. pornographic content. It was reported an American received 2200 pieces spam e-mail on average in 2002. Increasing by 2% per month, it will reach 3600 pieces spam e-mail in 2007. A survey by CNNIC found that every email user in China received 13.7 piece emails per week in 2004, including 7.9 piece spam emails. In America, spam emails cost enterprises up to 9 billions per year[1, 2]. A study reported that spam messages constituted approximately 60% of the incoming messages to a corporate network. Without appropriate counter-measures, the situation will become worse and spam email will eventually undermine the usability of email.

Anti-spam legal measures are gradually being adopted in many countries. In China, some experts advocated that an effective anti-spam e-mail measure should be carried out as early as possible. In 2003, AOL, Microsoft, Earthlink and Yahoo sued hundreds of marketing companies and individuals for sending deceptive spam using a new federal law called the CAN-SPAM Act, which prohibits such activities. But these legal measures have had a very limited effect so far due to Internet’s open architecture[3]. Hence, apart from legal measures, we should make use of some effective anti-spam e-mail technological approaches too. At present, most anti-spam e-mail approaches, which are too simple to stop spam e-mail efficiently, block spam messages by blacklist of frequent spammers.

We address the issue of anti-spam filtering with the aid of machine learning. Sahami et al trained a Naïve Bayesian classifier for anti-spam filtering, which learns to identify spam email after receiving training on messages that have been manually classified as spam or legitimate[4]. It was reported the method had impressive performance on unseen emails[5]. In this paper, we make use of an improved Bayesian-based approach to anti-spam. It is shown that the approach has better performance by our experimental results.

1 E-mail Corpus and Data Preprocessing

1.1 E-mail Corpus Collection

This project makes use of the test corpus from http://www.spamassassin.org/publiccorpus, which is an open available source. We select 1000 pieces e-mails randomly from the corpus according to spam e-mail ratio in China, including 580 spam e-mails, 420 legitimate e-mails. The corpus consists of all English emails whose attachments, html tags and email headers except the subject line have been stripped off.

1.2 Data Preprocessing

In our experiment, ith email in corpus is represented by $e_i$, a vector corresponding to $e_i$ is $x_p = (x_{1i}, x_{2i}, \ldots, x_{ni})$ where $x_{1i}$, $x_{2i}$, $\ldots$, $x_{ni}$ are the values of attributes $X_1$, $X_2$, $\ldots$, $X_N$.

Received 2004-06-16
In most previous anti-spam email filtering approaches, all attributes are binary, that is, \( x_i = 1 \) if some characteristic represented by \( X_i \) is present in the email \( e_i \); otherwise \( x_i = 0 \). The pattern simplifies the calculation, but doesn’t take account into attributes frequency. In our approach, we use vectors represented by attributes frequency. Frequency has two methods: absolute frequency and relative frequency. We select by attributes frequency. Frequency has two methods: absolute frequency and relative frequency. We select attributes based on word entropy. Word entropy is a measure of the information content of a word in the email classification context. It is calculated as follows:

\[
E(X_j) = \frac{S_j}{S} I(S_{\text{legit}, j}, S_{\text{spam}, j}) + \frac{1}{S} I(S_{\text{legit}, j+1}, S_{\text{spam}, j+1})
\]

where \( S_{x_j = 1} \) is the amount of email which contains word \( X_j \) or doesn’t contain word \( X_j \) in email sample corpus respectively. \( S_{\text{legit}, j = 0} \) is the amount of legitimate email which contains word \( X_j \) or doesn’t contain word \( X_j \) respectively. \( S_{\text{spam}, j = 1} \) is the amount of spam email which contains word \( X_j \) or doesn’t contain word \( X_j \) respectively. Therefore, word information gain is

\[
Gain(X_j) = I(S_{\text{legit}}, S_{\text{spam}}) - E(X_j)
\]

Attribute selection’s steps are as follows:
1) Except for words in stop-list (e.g. a, an, the, of and so on) and low-frequency low-information word, we stripe out all possible word from email corpus, then substitute each word by its base form (e.g. “earning” becomes “earn”) to avoid treating forms of the same word as different attributes. We adopt these methods which are referred to project file in our experiments[6].
2) Compute word information gain of every candidate word to email classification.
3) Words are sorted by word information gain, we find that in Eq.(4), \( I(S_{\text{legit}}, S_{\text{spam}}) \) is constant when training email corpus doesn’t vary, consequently, \( E(X_j) \) is smaller, \( Gain(X_j) \) is bigger. As a result, a sequence sorted by \( Gain(X_j) \) is equivalent to a sequence sorted by \( E(X_j) \).
4) We select words with the \( n \) highest information gain as attributes from the candidate words. It was tested that Bayesian classifier has the best effect when the dimension size is equal to 100[7], hence, we let the attribute size be 100 in our experiments.
5) Substitute all email in corpus by corresponding vectors.

\[\text{X}_3, \ldots, \text{X}_n \text{ of } e_i \text{ email respectively.}\]

\[\text{In most previous anti-spam email filtering approaches, all attributes are binary, that is, } x_i = 1 \text{ if some characteristic represented by } X_i \text{ is present in the email } e_i; \text{ otherwise } x_i = 0. \text{ The pattern simplifies the calculation, but doesn’t take account into attributes frequency. In our approach, we use vectors represented by attributes frequency. Frequency has two methods: absolute frequency and relative frequency. We select attributes based on word entropy. Word entropy is a measure of the information content of a word in the email classification context. It is calculated as follows:}\]

\[E(X_j) = \frac{S_{x_j = 1}}{S} I(S_{\text{legit}, j}, S_{\text{spam}, j}) + \frac{1}{S} I(S_{\text{legit}, j+1}, S_{\text{spam}, j+1})\]

where \( S_{x_j = 1} \) is the amount of email which contains word \( X_j \) or doesn’t contain word \( X_j \) in email sample corpus respectively. \( S_{\text{legit}, j = 0} \) is the amount of legitimate email which contains word \( X_j \) or doesn’t contain word \( X_j \) respectively. \( S_{\text{spam}, j = 1} \) is the amount of spam email which contains word \( X_j \) or doesn’t contain word \( X_j \) respectively. Therefore, word information gain is

\[Gain(X_j) = I(S_{\text{legit}}, S_{\text{spam}}) - E(X_j)\]

\[\text{Attribute selection’s steps are as follows:}\]
1) Except for words in stop-list (e.g. a, an, the, of and so on) and low-frequency low-information word, we stripe out all possible word from email corpus, then substitute each word by its base form (e.g. “earning” becomes “earn”) to avoid treating forms of the same word as different attributes. We adopt these methods which are referred to project file in our experiments[6].
2) Compute word information gain of every candidate word to email classification.
3) Words are sorted by word information gain, we find that in Eq.(4), \( I(S_{\text{legit}}, S_{\text{spam}}) \) is constant when training email corpus doesn’t vary, consequently, \( E(X_j) \) is smaller, \( Gain(X_j) \) is bigger. As a result, a sequence sorted by \( Gain(X_j) \) is equivalent to a sequence sorted by \( E(X_j) \).
4) We select words with the \( n \) highest information gain as attributes from the candidate words. It was tested that Bayesian classifier has the best effect when the dimension size is equal to 100[7], hence, we let the attribute size be 100 in our experiments.
5) Substitute all email in corpus by corresponding vectors.

\[\text{2 Anti-Spam Email Filter}\]

\[\text{2.1 Bayesian Classification}\]

\[\text{From Bayes’ theorem and the theorem of total probability, the probability that a email } e_i \text{ with vector } x_i = (x_1, x_2, \ldots, x_n) \text{ belongs to category } C_k \text{ (legitimate email is represented by } C_{\text{legit}} \text{, and spam email is represented by } C_{\text{spam}} \text{) is}\]

\[P(C_k | x_i) = \frac{P(C_k) P(x_i | C_k)}{\sum_{k \in \{\text{legit,spam}\}} P(C_k) P(x_i | C_k)}\]
In practice, the probability \( P(\mathbf{x}_e \mid C_i) \) is impossible to estimate without simplifying assumptions, because the possible values of \( \mathbf{x}_e \) are too many and there are also data sparseness problems, we assumes that \( X_1, X_2, \cdots, X_n \) are conditionally independent given the category \( C_i \). In other words, the probability \( P(\mathbf{x}_e \mid C_i) \) is just the product of the probabilities for individual attributes, which yields

\[
P(C_i \mid \mathbf{x}_e) = \frac{P(C_i) \prod_{k \in \{\text{legit, spam}\}} P(x_k \mid C_i)}{\sum_{C_i \in \{\text{legit, spam}\}} P(C_i) P(x_k \mid C_i)} \quad (6)
\]

A large number of empirical studies Refs.[8,9] have found the method to be surprisingly effective, despite the fact that the independence assumption is usually overly simplistic. As we use vectors represented by words frequency, we substitute it into Eq.(6) and yield

\[
P(C_i \mid \mathbf{x}_e) = \frac{P(C_i) \prod_{k \in \{\text{legit, spam}\}} P(x_k \mid C_i)^{N(x_k, e)}}{\sum_{C_i \in \{\text{legit, spam}\}} P(C_i) P(x_k \mid C_i)^{N(x_k, e)}} \quad (7)
\]

where \( P(C_i) \) may be obtained by estimating \( C_i \) category email ratio in total sample email.

\[
P(C_i) = \frac{S_{C_i}}{S} \quad (8)
\]

\( N(x_k, e) \) is frequency of word \( X_k \) which appears in email \( e \). \( P(X_k \mid C_i) \) is the probability of word \( X_k \) which appears in \( C_i \) category email, and

\[
P(X_k \mid C_i) = \frac{1 + \sum_{e \in \{\text{legit, spam}\}} N(X_k, e)}{n + \sum_{e \in \{\text{legit, spam}\}} N(X_k, e)} \quad (9)
\]

where \( n \) is vector size, \(|C_i|\) is \( C_i \) category email set.

Mistakenly blocking a legitimate mail (classifying a legitimate mail as spam) is generally more severe an error than letting a spam mail pass the filter (classifying a spam mail as legitimate). Let legit\(\rightarrow\)spam and spam\(\rightarrow\)legit denote the two error types. Invoking a decision-theoretic notion of cost, we assume that legit\(\rightarrow\)spam is \( \lambda \) times more costly than spam\(\rightarrow\)legit. An email is classified as spam if the following criterion is met

\[
P(C_{\text{spam}} \mid \mathbf{x}_e) \geq P(C_{\text{legit}} \mid \mathbf{x}_e) > \lambda \quad (10)
\]

### 2.2 Evaluation Criteria

Anti-spam email filter performance is often measured in terms of spam precision (SP) and spam recall (SR)

\[
SP = \frac{n_{\text{spam}} - \text{spam}}{n_{\text{spam}} - \text{spam} + n_{\text{legit}} - \text{spam}} \quad (11)
\]

where \( n_{\text{spam}} - \text{spam} \) are the numbers of spam emails classified as spam emails, \( n_{\text{legit}} - \text{spam} \) are the numbers of legitimate emails classified as spam emails mistakenly.

\[
SR = \frac{n_{\text{spam}} - \text{spam}}{N_{\text{spam}}} \quad (12)
\]

where \( N_{\text{spam}} \) are the total numbers of spam email, spam recall measures the percentage of spam email that the filter manages to block (intuitively its effectiveness), while spam precision measures the degree to which the blocked emails are indeed spam (the filter’s safety), which is more important factor in the performance evaluation. In our intuitiveness, it is difficult to compare the performance of different filters using spam recall and precision: each filter (or filter configuration) yields a pair of spam recall and precision results; without a single unifying measure. Therefore, we introduce a criterion \( F1 \), which incorporates spam precision and spam recall. It is defined as

\[
F1 = \frac{SP \times SR \times 2}{SP + SR} \quad (13)
\]

### 3 Experimental Results

10-fold cross-validation was used in our experiments, the email corpus \( S \) is partitioned randomly into ten subsets \( S_1, S_2, \cdots, S_{10} \) and the experiment is repeated ten times, each time reserving a different part for testing, and using the remaining nine parts for training. \( SP, SR \) and \( F1 \) are averaged over the ten iterations.

In our experiments, we compare effects of different types vector to filter, one is binary vector, the other is vector which is represented by word frequency. The result is shown in Tab.1. Meanwhile, we also compare different effects to different \( \lambda \) value, the experimental result is shown in Tab.2.

It is shown the approach based on word frequency is superior to the approach based binary vector in total performance from Tab.1, because the former provides
more information. In Tab.2, we learn that $\lambda$ is bigger, spam precision is higher, but spam recall decrease and $F_1$ decline a little too.

<table>
<thead>
<tr>
<th>Tab.1 Results using different type vectors</th>
</tr>
</thead>
<tbody>
<tr>
<td>Based on Binary</td>
</tr>
<tr>
<td>$SP$ (%)</td>
</tr>
<tr>
<td>96.82</td>
</tr>
<tr>
<td>$SR$ (%)</td>
</tr>
<tr>
<td>83.25</td>
</tr>
<tr>
<td>$F_1$ (%)</td>
</tr>
<tr>
<td>89.52</td>
</tr>
<tr>
<td>Based on word frequency</td>
</tr>
<tr>
<td>$SP$ (%)</td>
</tr>
<tr>
<td>98.65</td>
</tr>
<tr>
<td>$SR$ (%)</td>
</tr>
<tr>
<td>86.48</td>
</tr>
<tr>
<td>$F_1$ (%)</td>
</tr>
<tr>
<td>92.16</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Tab.2 Results using different $\lambda$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\lambda$ = 1</td>
</tr>
<tr>
<td>$SP$ (%)</td>
</tr>
<tr>
<td>98.65</td>
</tr>
<tr>
<td>$SR$ (%)</td>
</tr>
<tr>
<td>86.48</td>
</tr>
<tr>
<td>$F_1$ (%)</td>
</tr>
<tr>
<td>92.16</td>
</tr>
<tr>
<td>$\lambda$ = 9</td>
</tr>
<tr>
<td>$SP$ (%)</td>
</tr>
<tr>
<td>99.26</td>
</tr>
<tr>
<td>$SR$ (%)</td>
</tr>
<tr>
<td>84.59</td>
</tr>
<tr>
<td>$F_1$ (%)</td>
</tr>
<tr>
<td>91.34</td>
</tr>
</tbody>
</table>

4 Conclusions and Future Work

We introduce an improved Bayesian-based anti-spam approach. In our approach, we select attributes according to word entropy and use word frequency as vector and then find our filter improves total performances apparently. It is proved that our filter improves total performances apparently in our experiment.

Our future works will be:
1) At present, we select attribute corresponding to word in our experiments, we will consider attributes corresponding to phrase or non-textual contents in experiments.
2) Email sample sets in our experiments are English, we will study Chinese spam email filtering problem.
3) We will further employ other intelligent approaches on anti-spam email filter.

References

Brief Introduction to Author(s)
ZHAN Chuan (詹川) was born in 1973. He is now pursuing Ph.D degree in UESTC. His research interests include: information security, computer network, machine learning, data mining. E-mail: zhanchuan@uestc.edu.cn.
LU Xiang-liang (卢显良) was born in 1943. He is now a professor, and doctoral advisor in UESTC. His research interests include: operating system, computer network, information security. E-mail: xlu@uestc.edu.cn.
ZHOU Xu (周旭) was born in 1976. He is now pursuing the Ph.D degree in UESTC. His research interests include: distributed operation system, network storage. E-mail: xzhou@uestc.edu.cn.
HOU Meng-shu (侯孟书) was born in 1971. He is now pursuing the Ph.D degree in UESTC. His research interests include: network storage, P2P. E-mail: mshou@uestc.edu.cn.