ISSN 1870-4069

A Comparative Study on the Effectiveness of Naïve Bayes Classifiers in Spam Filtering

Sara Besharati, Seyed Habib Hosseini Saravani, Alexander Gelbukh

Instituto Politécnico Nacional,
Centro de investigación en Computación,
Mexico

{besharatisara62, hosseinihaamed}@gmail.com,
gelbukh@gelbukh.com

Abstract. Naïve Bayes classifiers are among the very effective machine learning classifiers being used for the task of spam filtering. However, there are different kinds of Naïve Bayes classifiers based on their training algorithms and also their attitudes toward the data distribution and representation. In this research, conducting a comparative study on the effectiveness of multinomial, Bernoulli, and Gaussian Naïve Bayes classifiers in spam filtering, we show that multinomial and Bernoulli Naïve Bayes algorithms are more effective than Gaussian Naïve Bayes algorithm for the task of spam filtering. However, based on F1 Score, multinomial Naïve Bayes has the best performance among all the three models.

Keywords: Email, spam, Naïve bayes, multinomial, Bernoulli, gaussian.

1 Introduction

The tendency to use email technology as a kind of medium has increased to a large extent since internet became an inevitable aspect of everyday life of the majority of people in the world. Consequently, little by little, the ones who wanted to advertise their products and services, started making use of this phenomenon (email) to absorb their customers. Today, we receive many unwanted emails that occupy the space of our inboxes, and we have to spend some of our time removing them. Thus, the need for having a system that can filter the emails in order to detect the unwelcome mass emails being sent to the users (spam emails) has become more essential than before. Spam emails target their users mainly to advertise their contents; nevertheless, in some cases, they have destructive and even criminal intentions. Thus, the spam filtering systems should be strong and accurate enough to distinguish between spam emails and non-spam ones effectively.

Among all the machine learning method being used for the task of spam detection, Naïve Bayes classifiers are very famous. In this research, we compare the performance of three main Naïve Bayes algorithms – multinomial, Bernoulli, and Gaussian Naïve Bayes classifiers – that can be used for task of spam filtering. We compare the effectiveness of these classifiers in distinguishing spam emails from non-spam emails and introduce the most effective ones for this task.

In the following paragraphs, we review the related work in Section 2; we explain the algorithms of the three Naïve Bayes methods in Section 3; we compare the results obtained from the models developed in this research in Section 4; and we summarize our findings and talk about future work in Section 5.

2 Related Work

Schneider [1] applied multivariate Bernoulli Naïve Bayes and multinomial Naïve Bayes classifiers to the task of spam detection and reported that the multinomial models achieved higher accuracy than the Bernoulli model did. The multinomial Naïve Bayes and the Bernoulli model in that research were reported to have the accuracy 98.86% and 98.00% respectively, which is quite high.

McCue [2] compared the accuracy of Support Vector Machine (SVM) and Bayesian classifiers for the task of spam detection. The data used in [2] consisted of a matrix with 2000 email rows, each with 2000 feature columns, resulting in a 2000 × 2001 matrix. Thus, each email in this dataset has 2000 features, each of which is a binary of a word's existence within that email. The results of that paper showed that, despite the simplicity of Naïve Bayes algorithm, it can give a better prediction results on the testing set used in that research, coming in at a respectable 97.8% in comparison with the best accuracy obtained from SVM classifiers, which was 96.6%. Also, that paper reported that Gaussian Naïve Bayes algorithm cannot work with spam detection because of its 70% accuracy.

Méndez et al. [3] applied four different Naïve Bayes classifiers to the spam classification task. They presented a comparative study for the impact of five feature selection methods when using four variants of the original Naïve Bayes algorithm working as spam filter. The feature selection methods studied were Information Gain (IG), Odds ratio (OR), Document Frequency (DF) (*d*) $\chi 2$ statistic, and Mutual Information (MI). Moreover, we have analyzed the following Naïve Bayes alternatives: (*i*) Multivariate Bernoulli, (*ii*) Multinomial Naïve Bayes, (*iii*) Multivariate Gaussian, and (*iv*) Flexible Bayes. That research reported that, considering DF as the feature extraction, first Bernoulli and then multinomial Naïve Bayes models had the best performance, but the OR method presents a high performance level when it is used with Gaussian-based Naïve Bayes algorithms.

Almeida et al. [4] performed a comparison of performance achieved by four Naïve Bayes anti-spam filters – multinomial term frequency Naïve Bayes, multinomial Boolean Naïve Bayes, multivariate Bernoulli Naïve Bayes, Flexible Bayes – to classify messages as legitimate or spam. Among all the classifiers they used, multivariate Bernoulli Naïve Bayes achieved the best performance having the accuracy 98.90%. The next best accuracy, which was 97.47%, was obtained by the multinomial Boolean Naïve Bayes in that research.

In this research, doing a comparative research, we focus on the aspects which were not paid enough attention in the previous researches. We analyze the effectiveness of multinomial, Bernoulli, and Gaussian Naïve Bayes classifiers in filtering spam emails, and we show that each of these three classifiers can be effective for a specific task in spam filtering.

Table 1. Number of positive (spam) and negative (non-spam) emails in the dataset.

Positive	1368
Negative	4360

3 Methodology

We use a Kaggle dataset named 'Spam filter' [5], which consists of 5528 emails -- 4260 non-spam and 1368 spam emails (Table 1). We randomly chose 15% of the data and allocated it for the test dataset, and the rest of the data was used as the training dataset. Also, for the implementation of the codes and classification of the data, we use Scikit-learn [6], which is a free software machine learning library for the Python programming language.

3.1 Naïve Bayes

Naïve Bayes algorithm is a kind of a frequently used supervised learning method that examines all its training input and applies Bayes theorem with the "naïve" assumption of conditional independence between features given the value of the class variable [6]. Equation 1 below shows Bayes theorem, where *c* stands for class variable and x_1 through x_n are dependent feature vectors:

$$P(C \mid x_1, \cdots x_n) = \frac{P(C)P(x_1, \ldots, x_n|C)}{p(x_1, \ldots, x_n)}. \quad (1)$$

There are different kinds of Naïve Bayes classifiers based on their training and classification algorithms and their attitude toward the data distribution. In the following of this section, we will review the training algorithm of multinomial, Bernoulli, and Gaussian Naïve Bayes classifiers.

3.2 Multinomial Naïve Bayes

This algorithm is implemented to the data that is multinomially distributed and is one of the Bayes variants that is usually used in text classification [6]. The distribution is parametrized by vectors $\theta_y = (\theta_{y1}, \ldots, \theta_{yn})$ for each class y, where *n* is the number of features – the size of the vocabulary – and θ_{yi} is the probability $P(x_i|y)$ of feature *i* appearing in a sample belonging to class *y*. The parameters θ_y is estimated by a smoothed version of maximum likelihood, i.e. relative frequency counting:

$$\hat{\theta}_{yi} = \frac{N_{yi} + \alpha}{N_y + \alpha n} . \quad (2)$$

In the equation above, $N_{yi} = \sum_{x \in T} x_i$ is the number of times feature *i* appears in a sample of class y in the training set T, and $N_y = \sum_{i=1}^{n} N_{yi}$ is the total count of all features for class y. The smoothing priors $\alpha \geq 0$ accounts for features not present in the learning

samples and prevents zero probabilities in further computations. Setting $\alpha = 1$ is called Laplace smoothing, while $\alpha < 1$ is called Lidstone smoothing [6].

3.3 Bernoulli Naïve Bayes

When the data is distributed according to multivariate Bernoulli distributions, where there may be multiple features but each one is assumed to be a binary-valued (Bernoulli, Boolean) variable, Bernoulli Naïve Bayes can be used for the classification of the data [6]. In this kind of classification, this class requires samples to be represented as binary-valued feature vectors. Bernoulli Naïve Bayes makes the decision based on Equation 3, where P(I | y) refers to the probability of finding a term *i* in a given message belonging to class *y*:

$$P(x_i / y) = P(i / y)\, x_i + (1 - P(i / y))\, (1 - x_i). \quad (3)$$

This learning differs from multinomial Naïve Bayes rule in that, unlike the multinomial variant, it does not simply ignore feature *i* if it does not occur in class *y*. In the case of text classification, word occurrence vectors may be used to train the model. Bernoulli might perform better on some datasets, especially those with shorter documents [6].

3.4 Gaussian Naïve Bayes

In comparison with multinomial Naïve Bayes, Gaussian Naïve Bayes classifier assumes that the distribution associated to each term is a Gaussian distribution for each class *y*, and considers that the values of the attributes are independent in each class. Gaussian Naïve Bayes classifier uses continuous features by representing the frequency of the terms in an input [6, 2]. The likelihood of the features in this kind of classification is assumed to be Gaussian:

$$P(x_i \mid y) = \frac{1}{\sqrt{2\pi\sigma_y^2}} \exp\left(-\frac{(x_i-\mu_y)^2}{2\sigma_y^2}\right). \quad (4)$$

In Equation 4, μ_y and σ_y represent the mean and the standard deviation of the appearance frequency of the terms in the inputs belonging to class *y* [6, 2].

3.5 Evaluation

Since accuracy is the most intuitive metric that can simply give us a ratio of correctly predicted observation to the total observations, the first evaluation metric we used was accuracy. In addition, in order to have the ratio of correctly predicted positive observations to the total predicted positive observations and also the ratio of correctly predicted positive observations to the all observations in actual class, we used the metrics precision and recall respectively. Finally, to take both false positives and false negatives into account, we used F1 Score, which is the weighted average of Precision and Recall:

Table 2. Comparison on different Naïve Bayes models used in this research.

Models	Precision	Recall	F1 Score
Multinomial Naïve Bayes	99.51	96.72	98.10
Bernoulli Naïve Bayes	95.67	98.02	96.83
Gaussian Naïve Bayes	76.92	85.10	80.80

Table 3. Comparison of our models – Multinomial Naïve Bayes (MNB), Bernoulli Naïve Bayes (BNB), Gaussian Naïve Bayes (GNB) – with the models in previous works.

Models	Accuracy (%)
MNB	99.06
BNB	98.48
GNB	91.16
MNB - Schneider	98.86
BNB - Schneider	98.00
MNB - Almeida	97.47
BNB - Almeida	98.90
GNB - McCue	70.00

$$\text{Accuracy} = \frac{\text{TP+TN}}{\text{TP+FP+FN+TN}}, \tag{5}$$

$$\text{Precision} = \frac{\text{TP}}{TP+FP}, \tag{6}$$

$$\text{Recall} = \frac{TP}{TP+FN}, \tag{7}$$

$$\text{F1 Score} = \frac{2\times(\text{Recall} \times \text{Precision})}{(\text{Recall} + \text{Precision})}. \tag{8}$$

Equations 5, 6, 7, and 8 show the formulae for the calculation of accuracy, precision, recall, and F1 Score respectively, where TP (True Positive), TN (True Negative), FP (False Positive), and FN (False Negative) refer to the correctly predicted spam emails, correctly predicted non-spam emails, incorrectly predicted spam emails, and incorrectly predicted non-spam emails.

4 Experimental Results

As Tables 2 and 3 shows, the multinomial and the Bernoulli Naïve Bayes have had a very good performance in classification of the emails. In this research, the multinomial models had the best performance reaching the accuracy 99.06%, and after that the

Table 4. Confusion Matrix of multinomial NB model.

	Positive	**Negative**
Positive	207	7
Negative	1	645

Table 5. Confusion Matrix of Bernoulli NB model.

	Positive	**Negative**
Positive	199	4
Negative	9	648

Table 6. Confusion Matrix of Gaussian NB model.

	Positive	**Negative**
Positive	160	28
Negative	48	624

Bernoulli model had the accuracy 98.48%. Also, it can be seen in Tables 2 and 3 that Gaussian Naïve Bayes model cannot compete with the multinomial and the Bernoulli Naïve Bayes models in spam filtering having the accuracy 91.16%.

As the confusion matrices of the three models (Tables 4, 5 and 6) show, the multinomial Naïve Bayes model has the best precision among all the three models, detecting the most number of spam emails correctly. However, when it comes to recall, the Bernoulli Naïve Bayes has the best performance, which means that this model sends the least number of spams to your inbox, but, comparing with multinomial model, it has more mistakes in spamming your non-spam emails. Figure 1 below shows the differences between the models considering different metrics for the evaluation of the models.

5 Conclusions and Future Work

The results obtained from the models developed in this research showed that, among the three well-known types of Naïve Bayes classifiers, the multinomial and the Bernoulli classifiers are more appropriate for the task of spam filtering. Both multinomial and Bernoulli algorithms had a good performance in detecting the spam emails, and their results were very close to each other; however, the multinomial model was stronger than the Bernoulli model considering precision as the evaluation metric, while the Bernoulli model had a better recall than the multinomial Naïve Bayes model. Nevertheless, it must be borne in mind that non-spam emails are more important than spam emails for the users, and a spam filtering system should make the less possible mistakes in classifying non-spam emails. As a result, if we consider precision as the evaluation metric, our results show that the multinomial Naïve Bayes model has the best performance among all the models we developed in this research.

For the future work, working on a selective Bayes classifier that can do an effective feature extraction for the task of spam filtering is aimed.

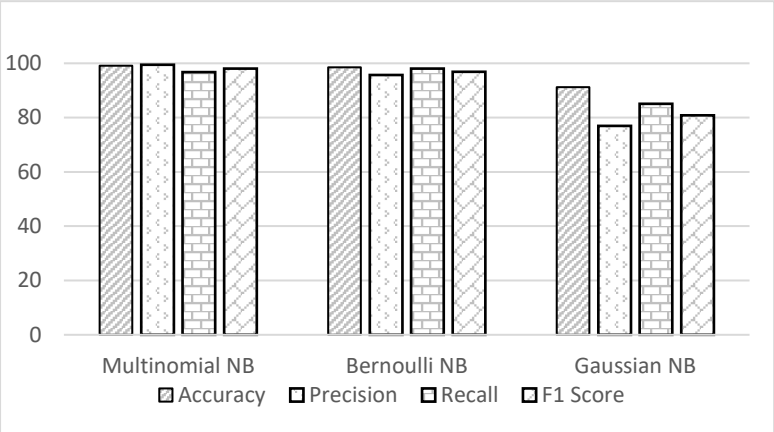

Fig. 1. Comparison of the results of different Naïve Bayes models used in this research.

References

1. Schneider, K.M.: A comparison of event models for Naïve Bayes anti-spam e-mail filtering. In: Proceedings of the tenth conference on European chapter of the Association for Computational Linguistics, Association for Computational Linguistics, 1, pp. 307–314 (2003)
2. McCue, R.: A comparison of the accuracy of support vector machine and Naïve Bayes algorithms in spam classification. University of California at Santa Cruz (2009)
3. Méndez, J.R., Cid, I., González-Peña, D., Rocha, M., Fernandez-Riverola, F.: A comparative impact study of attribute selection techniques on Naïve Bayes spam filters. In: ICDM´08 Industrial Conference on Data Mining, pp. 213–227 (2008)
4. Almeida, T.A., Yamakami, A., Almeida, J.: Probabilistic anti-spam filtering with dimensionality reduction. In: SAC´10 Proceedings of the ACM Symposium on Applied Computing, pp. 1802–1806 (2010)
5. Karthickveerakumar: Spam filter. 1 (2017)
6. Scikit-learn: 1.9. Naïve Bayes (2020)