Proceedings of MSS 2025 Symposium: Automation and Smart Technologies in Petroleum Engineering
DOI: 10.54254/2755-2721/142/2025.KL21549

A Review of Bayes Machine Learning for Spam Filtering Applications

Zehao Song[1,a,*]

[1]*School of Mathematics, Shanghai University of Finance and Economics, Shanghai, 200433, China*
a. songzehao111@stu.sufe.edu.cn
**corresponding author*

Abstract: The Naive Bayes algorithm uses the theorem of Bayes to filter spam emails, achieving good filtering results. The improved Bayes algorithm addresses the assumption of "feature independence given the class" in Naive Bayes algorithm, allowing for a broader application range. This paper reviews the main content and representative achievements of both the Naive Bayes algorithm and the improved Bayes algorithm, and analyzes the advantages and disadvantages of each method. This study finds that the Naive Bayes algorithm has a limited application range due to the assumption of "feature independence given the class" while the improved Bayes algorithm effectively solves this problem and it has better applicability. This paper aims to help researchers engaged in spam filtering better understand and leverage the potential of the theorem of Bayes in spam filtering, providing a summary reference to promote technological innovation in related fields and better problem-solving, as well as facilitating the understanding of other readers and the application of Bayes filtering methods.

Keywords: Bayes, machine learning, spam, filtering

1. Introduction

Learning is a practical process that involves acquiring new knowledge, values, skills, and so on. Machine learning is an interdisciplinary field that belongs to artificial intelligence. It allows computers to mimic human thinking and learning, to reflect and summarize, and ultimately to solve the same or similar problems successfully.

With the popularization of the Internet, email has become an important tool for people's daily communication. However, the rampant spam is seriously affecting our normal life and communication. Spam refers to unsolicited, automatically sent and valueless or potentially harmful emails. Spam includes unwanted messages such as commercial advertisements, pornography, violence, and viruses, which occupy system memory and increase the time people spend processing information. According to a survey, 40% of emails in China are spam, and 90% of emails in the United States are spam [1]. Moreover, spam has also caused serious economic losses to China. Therefore, it is very important to filter spam. At present, there are mainly black-and-white list filtering methods, rule-based filtering and probability-based filtering, but these methods all have obvious shortcomings [2].

The current stage of research mainly focuses on finding more efficient and accurate filtering methods, and there are relatively few review articles that compare and summarize current research

DOI: 10.54254/2755-2721/142/2025.KL21549

results. Based on this, this article aims to introduce the Naive Bayes algorithm and improved Bayes algorithms, analyze their principles and compare their advantages and disadvantages, to better address the problem.

2. Naive Bayes Algorithm

A naive Bayes algorithm is a probability-based classification algorithm based on The theorem of Bayes, which originates from a Bayes classifier. The theorem of Bayes was proposed by 18th-century mathematician Thomas Bayes, and this theorem is used to describe the probability of an event occurring given a certain condition. The Bayes algorithm uses probabilities to describe the relationship between classes and features, and determines classification by calculating the posterior probability. Therefore, the naive Bayes algorithm is also known as a probability-based classification [3].

2.1. The Theorem of Bayes

The mathematical expression of the theorem of Bayes is:

$$P(C|X) = \frac{P(X|C)\times P(C)}{P(X)} \tag{1}$$

where:$P(X)$ is the total probability of the feature X, which can be omitted when it is larger because it is known to be constant in that probability calculation.

$P(C|X)$ is the conditional probability of feature C given the known category X occurring.

$P(X|C)$ is the conditional probability of feature X occurring given the known category C

$P(C)$ is the prior probability of category C, that is, the probability of category C occurring. Naive Bayes method calculates the posterior probability based on The theorem of Bayes, using the prior probability to determine the posterior probability when a certain event $X(x_1 \ldots x_n)$occurs, selecting features $C(C_1 \ldots C_n)$ that maximize $P(C|X)$ to achieve the largest C_k as the classification result.

2.2. Naive Bayes Mail Classifier

X_i=1/0 indicates that the feature is present/not present in the mail, and C=1/0 means the email is/is not spam. The core idea of the Plain Bayes algorithm is based on the theorem of Bayes, which categorizes emails by calculating the posterior probability of each category.

The general steps of the Bayes algorithm for filtering spam emails are: Data collection and processing, feature selection (Each word in the email is treated as a feature X_i, where X_i=1/0 indicates the presence/absence of the feature in the email), training (calculating the probability of each word appearing in spam and non-spam emails), applied to real-world problems.

The training phase in turn includes:

(1) Calculate the prior probability $P(C)$. That is, the proportion of spam in the training set,

$$P(C = 1) = \frac{\text{Number of spam samples}}{\text{Total sample size}} \tag{2}$$

$$P(C = 0) = 1 - P(C = 1) \tag{3}$$

(2) Calculate conditional probabilities $P(X_j = x_j|C)$. For each category C and each feature x_j, count the frequency of the feature appearing in that category and calculate its conditional probability

$$P(X_j = x_j|C = 1) = \frac{\text{The number of occurrences of feature } x_j \text{ in spam emails}}{\text{Total number of occurrences of all features in spam emails}} \tag{4}$$

$$P(X_j = x_j|C = 0) = \frac{\text{The number of occurrences of feature } x_j \text{ in non-spam emails}}{\text{Total number of occurrences of all features in non-spam emails}} \tag{5}$$

DOI: 10.54254/2755-2721/142/2025.KL21549

(3) Detection. For each new email in the test set, compute the posterior probability that it belongs to spam and non-spam, select the category with the highest probability as the result and test the accuracy.

To simplify the computation, this algorithm requires that the n components of the feature vector $X(x_1 \ldots x_n)$ are mutually independent, that is

$$P(X = x|C) = \prod_{i=1}^{n} P(X_i = x_i \mid C) \quad (6)$$

Based on the fact that P(X=x) in the denominator of the Bayes formula is constant for the feature C and can be omitted when it is relatively large, this way the posterior probability can be written as follows:

$$P(C|X) \propto P(C) \prod_{i=1}^{n} P(X_i = x_i \mid C) \quad (7)$$

For the spam classification problem, we calculate and compare the sizes of (8) and (9) respectively,

$$P(C = 1|X = x) = \prod_{i=1}^{n} P(X_i = x_i \mid C = 1)P(C = 1) \quad (8)$$

$$P(C = 0|X = x) = \prod_{i=1}^{n} P(X_i = x_i \mid C = 0)P(C = 0) \quad (9)$$

If the result of (8) is greater than (9), it is classified as spam; otherwise, it is not spam.

2.3. Problems with the Naive Bayes Algorithm

(1) Underflow problem. This refers to the situation when naive Bayes calculates joint probabilities by multiplying the conditional probabilities of each feature, which are often very small. When the amount of data is large, the continuous multiplication in equations (8) and (9) can cause the result to approach zero, and in Python, this will default to zero during computation, making it impossible to compare sizes. To address this, the conditional probabilities in equation (6) can be logarithmically transformed before calculation.

(2) The conditional probability is zero. This indicates that the feature does not appear in the email, and a certain term $P(X_j = x_j|C)$ in equation (6) is zero, making equations (8) or (9) zero. To avoid the situation where the probability is zero, Laplace smoothing is usually used to smooth the estimated probabilities.

It can be made that $P(X_j = x_j|C) = \frac{\text{Number of occurrences of feature x\_j in category C} + 1}{\text{Total number of occurrences of all features in category C} + V}$, where V is the size of the feature space, which is usually the number of all possible features in the training set. In this way, such cases can be effectively avoided.

Although the Naive Bayes algorithm is simple and intuitive, it has high accuracy and speed when dealing with large types of data, and can even be compared to neural networks and decision tree algorithms. However, it requires that the feature vectors $X(x_1 \ldots x_n)$ are mutually independent when the classification C is known, which is often difficult to achieve in general cases and may lead to unsatisfactory classification results [4].

3. Improved Bayes Algorithm

As mentioned above, the "feature independence" requirement of Naive Bayes algorithm is generally not satisfied in practice. Certain features in emails may have strong correlations. Therefore, researchers have proposed a variety of improved algorithms to relax or adjust this independence assumption. Here are some of the main approaches:

DOI: 10.54254/2755-2721/142/2025.KL21549

3.1. A Semi-naive Bayes Algorithm

3.1.1. TAN Algorithm

In real spam emails, some features may have a strong correlation, and ignoring these correlations may lead to misjudgments by the model, affecting classification results. For example, the words "free" and "offer" often appear together in certain spam emails. The naive Bayes algorithm would consider them independent, but in reality, "free" and "offer" have a strong correlation: when they appear together, the probability of the email being spam is higher than when they appear separately.

The enhanced Naive Bayes classifier improves upon the basic Naive Bayes classifier and effectively addresses the assumption of "features being mutually independent for classification." The enhanced Naive Bayes classifier utilizes the concept of "Markov blanket" to introduce the dependencies between features into the model, relaxing the assumption of feature conditional independence, while directly strengthening it while maintaining the basic structure of the Naive Bayes classifier. The Markov blanket indicates that the dependencies between features are local and can be represented by establishing an appropriate network structure. Typically, such dependencies can be represented by graphical models, and the simplest enhanced Naive Bayes classification algorithm is the Tree-Augmented Naive Bayes (TAN) model, which allows features to be connected through a tree structure rather than being completely independent, with class nodes and attribute nodes connected through parent-child relationships.

By modifying the CL algorithm of Chow et al, Friedman proposed a learning algorithm for Tree Augmented Naive Bayes (TAN) classifier [5]. The main steps of TAN are as follows:

a. Calculate the mutual information between each pair of features to measure the correlation between features. The mutual information formula is:

$$I(x_i, x_j|C) = \sum_{x_i,x_j,C} P(x_i, x_j, C) \log \frac{P(x_i,x_j|C)}{P(x_i|C)P(x_j|C)} \tag{10}$$

Among them, the greater the mutual information, the stronger the correlation between features x_i and x_j under the condition of category C.

b. Based on the results of mutual information calculation, construct the maximum spanning tree between features. In the tree, each feature depends on at most one other feature.

c. Add the dependency structure of the spanning tree to the plain Bayes model. Conditional dependencies between features are considered when calculating P(X|C):

$$P(X|C) = P(X_1|C) \prod_{i=2}^{n} P(x_i \mid x_{parent(i)}, C) \tag{11}$$

Here, parent(i) denotes the parent feature of feature x_i.

d. Classify using the naive Bayes method as mentioned in the previous text.

The advantage of Tree-Augmented Naive Bayes (TAN) lies in relaxing the assumption of feature conditional independence, allowing it to capture dependencies between features and making it applicable to a wider range of scenarios; at the same time, TAN maintains a lower computational complexity through its tree structure, making it suitable for handling large-scale data. However, TAN only establishes tree-structured dependencies and cannot handle more complex multivariate dependencies, and the computation of generating the tree increases training time, which may lead to poor performance when dealing with data that has more complex dependencies.

3.1.2. Network Reliability Enhancement Algorithm

Yongkang Xing et al. proposed a general belief network classifier, GBNC, which directly uses the belief network established on the learning database as a classifier [6]. The general belief network

DOI: 10.54254/2755-2721/142/2025.KL21549

classifier is a belief network-based classification method that automatically constructs a belief network in the training data for classification. Its construction process uses an independence test algorithm to identify dependency relationships between variables from the data and generates a belief network structure. In the belief network, nodes represent variables, and edges represent conditional dependencies. In this way, GBNC can dynamically capture the dependency relationships between variables.

During the classification stage, GBNC uses the attribute values of the instances to be classified as evidence to input the belief network. Then, the probability of class variables is calculated by the reasoning algorithm. The category with the highest probability is the classification result of the instance. The advantage of the GBNC classifier is that it can automatically establish dependencies according to data, flexibly deal with complex data and interdependent variables. However, compared with TAN, both the learning time and complexity of its algorithm are longer and higher.

3.2. Effective Knowledge Learning Email Filtering Algorithm

3.2.1. Spam Filtering Algorithm Based on Email Filtering

Rennie built an email filtering algorithm that integrates machine learning models and a knowledge base based on the Naive Bayes algorithm. He named this algorithm [7].

Email filtering spam filters use Bayes algorithms to initially classify emails, identifying spam and non-spam. When the system makes misclassifications, users can manually correct them, and the system uses this feedback to retrain the model, continuously updating and optimizing its classification capabilities.

Compared with the Naive Bayes algorithm, the spam filtering-based spam algorithm can automatically improve its recognition ability through continuous user feedback and retraining. However, this algorithm mainly relies on simple additions and subtractions of mail feature information and lacks more complex and effective learning methods, which means that the system is still prone to misjudgment for some complex and hidden spam emails. At the same time, the learning mechanism of ifile is relatively simple, which limits the filtering effect and robustness of the system.

3.2.2. A Bayes Spam Filtering Algorithm Based on Minimum Risk

Lin et al. and Cai et al. have improved the naive Bayes algorithm and proposed a Bayes email classification algorithm based on minimum risk [8-9]. The algorithm based on minimum risk introduces a risk function in decision-making, which is used to measure the cost of misclassification and to choose the optimal classification strategy on this basis. In the classification of this algorithm, when an email is classified as spam or non-spam, it not only considers the probability of the email belonging to each category, but also considers the "risk" that each misclassification may bring. This "risk" is often expressed by a cost matrix. The algorithm based on minimum risk chooses the classification strategy that minimizes the total risk by integrating the costs of all errors. However, this algorithm does not have learning ability compared with other algorithms, so its application degree is not high.

3.2.3. Spam Filtering Algorithm Based on Bayes Neural Network

Traditional neural networks are typically deterministic and cannot adequately express the uncertainty of model parameters. To address this issue, Hui-juan Li et al introduced Bayes inference to neural networks, enabling the model not only to output classification results but also to provide uncertain information about the classification [10]. This type of neural network that combines Bayes methods can adjust the network's weights through Bayes inference, thus considering the uncertainty of

DOI: 10.54254/2755-2721/142/2025.KL21549

different parameters during training, effectively avoiding overfitting, enhancing the model's generalization ability, and making it more robust when encountering new types of spam emails. In addition, Bayes neural networks can incorporate prior knowledge, making them superior to naive Bayes algorithms in both classification accuracy and efficiency.

4. Conclusion

The Naive Bayes algorithm is simple and efficient but has high requirements for feature independence, so its application scope is limited. The improved Bayes algorithm effectively solves this problem and is therefore more widely used. However, as filtering technology continues to improve, spam is also evolving. Spam emails with images, special symbols, and malicious websites are a major challenge we face. This paper does not discuss these issues in depth. In addition, further research is needed on spam filtering in terms of improving classifier generalization ability, reducing energy consumption during large-scale data processing, and protecting user privacy, in order to cope with the increasingly complex email environment and user needs.

References

[1] *Qin Zhiguang, Luo Qin, Zhang Fengli. Research on a Hybrid Spam Filtering Algorithm [J]. Journal of University of Electronic Science and Technology of China, 2007, (03): 485-488.*

[2] *Fan Shilun, Xue Tianjun, Xia Wei. Design and Implementation of a Spam Filtering System Based on Bayesian Algorithm and Fisher Algorithm [J]. Information Network Security, 2012, (09): 18-22.*

[3] *Peng Ge. A Review of Research on Naive Bayes Algorithm in Spam Filtering [J]. Computer Knowledge and Technology, 2020, 16 (14): 244-245+247. DOI:10.14004/j.cnki.ckt.2020.1577*

[4] *Zhao Zhiguo, Tan Minsheng, Li Zhimin. A Review of Improved Bayesian Spam Filtering Algorithms [J]. Journal of Nanhua University (Natural Science Edition), 2006, (01): 33-38. DOI:10.19431/j.cnki.1673-0062.2006.01.009.*

[5] *Friedman, Goldszmidt. Building Classifiers using Bayesian Networks [A]. Proceedings of National Conference on Artificial Intelligence [C]. Menlo Park: CA: AAAI Press, 1996.*

[6] *Xing Yongkang, Shen Yidong. Credibility Network Classifier [J]. Journal of Chongqing University (Natural Science Edition), 2000, 23(5): 49-52.*

[7] *Rennie J. iFile: An application of machine learning to e-mail filtering[C]. Boston: Proceedings of KDD-2000 Text Mining Workshop, 2000.Rennie J. iFile*

[8] *Lin Y P, Chen Z P, Yang X L, et al. Mail filtering based on the risk minimization Bayesian algorithm. 6th World Multiconference on Systemics, Cybernetics and Informatics (SCI 2002)[J]. Proceedings-Industrial Systems and Engineering III, 2002, 17(3): 282-285.*

[9] *Cai Lijun, Shi Ronghua. Design of a New Email Filtering System Model [J]. Computer Engineering, 2003, 29(16): 167-169.*

[10] *Li Huijuan, Gao Feng, Guan Xiaohong, et al. A Spam Filtering Method Based on Bayesian Neural Networks [J]. Microelectronics and Computer, 2005, 22 (4): 107-111.*