

Various Machine Learning Models for Classifying Trash Email

Baiyue PETER Zhao *

The Affiliated High School of Peking University, Beijing, China

* Corresponding Author Email: author@abcd.edu

Abstract. Unsolicited or trash emails have become a major issue in modern digital communication, creating significant inconvenience and potential risks for users. These emails not only waste time and resources but may also contain malicious links, phishing attempts, or harmful attachments. As a result, spam detection has become an important area of research, and Artificial Intelligence (AI) models are increasingly applied to improve filtering accuracy. In this project, several machine learning models were employed to classify the spam and legitimate emails. The dataset was preprocessed, and the classification results were not only analyzed in text form but also visualized through charts to provide a clearer comparison of performance. Different algorithms, including traditional classifiers and advanced models, were tested and evaluated. The results demonstrate that AI-based approaches can achieve high accuracy and efficiency in identifying spam emails, highlighting their effectiveness as reliable tools for addressing the persistent problem of unwanted email communication.

Keywords: Machine learning, spam email, artificial intelligence.

1. Introduction

Email spam is the term for unsolicited messages delivered in large quantities over email. Spam has increased dramatically since the early 1990s; estimates indicate that by 2014, it accounted for around 90% of all email traffic worldwide. The recipient of spam may have to handle, filter, or remove these unsolicited messages, which is a burden. Spam is an example of a "negative externality"—a consequence of an action that impacts those who were not involved in the decision-making process—because it costs recipients money without providing recompense to the sender. Although different jurisdictions have different legal definitions and statuses for spam, laws and lawsuits have not been able to effectively curb spam. The majority of spam emails are business-related. Regardless of whether they are commercial or not, a lot of them are not just harmful because they might have links that take users to malicious websites or phishing sites, or they might contain malware as file attachments. Other spammers occasionally purchase these gathered email addresses.

Spammers may intentionally commit fraud in order to distribute their messages. Spammers frequently create "disposable" accounts at different ISPs using fictitious names and other contact details. Additionally, they frequently pay for these accounts using stolen or fictitious credit card numbers. This enables them to swiftly switch between accounts when the host ISPs find and terminate each one.

Senders may take extreme measures to hide where their messages are coming from. Big businesses may contract with a different company to deliver their messages, putting the blame for complaints or email blockage on a third party. Some attackers exploit email address spoofing, which is even simpler than IP spoofing. Since the Simple Mail Transfer Protocol (SMTP) does not include built-in authentication, spammers can disguise messages as though they were sent from any address. To counter this, many ISPs and domains now require SMTP-AUTH, which enforces account verification and ensures that the actual sender can be positively identified.

Because the receiving mail server logs the real connection from the previous mail server's IP address, senders are unable to fully spoof email delivery chains (the 'Received' header). Some spammers try to combat this by forging extra delivery headers that give the impression that the email has already passed through numerous trustworthy servers.

For authentic email users, spoofing can have detrimental effects. They risk being incorrectly labeled as spammers, in addition to having their email inboxes overflowing with "undeliverable" emails.

There are lots of similar studies in this topic like detection of spam email [1], or use a few models to detect spam email [1]. In the process this study works in this project, it can be found two problems such as whether AI model is efficiency, and which is better [2]. This paper is also compare different models like SVM, BERT, DistilBERT, RoBERTa and XLNet [3]. This paper discusses how to compare different models based on different elements [4].

2. Method

2.1. Dataset Preparation

This study used the dataset called 20newsgroups [5]. The purpose of this project is to compare the effects in different model, and choose the best.

2.2. Machine Learning Models

2.2.1 K-Nearest Neighbors

One non-parametric supervised learning technique is the k-nearest neighbors algorithm (k-NN). Joseph Hodges and Evelyn Fix created it initially in 1951, and Thomas Cover later added to it. As a k-NN classifier, which produces a class membership as its output, it is most frequently employed for classification. By a plurality vote of its neighbors, an item is classified into the class that is most prevalent among its k nearest neighbors (k is a positive number, usually small). The object is merely put into the class of that one nearest neighbor if $k = 1$.

2.2.2 Support Vector Machines

Support Vector Machines (SVMs) are supervised learning models that aim to identify an optimal boundary for classification and prediction tasks. At their core, SVMs are max-margin classifiers, meaning they attempt to find the hyperplane that separates different classes of data with the maximum possible margin. This margin maximization principle reduces the risk of misclassification and often improves the model's generalization ability when applied to unseen data.

The learning process behind SVMs involves transforming the input space using kernel functions, which enable the algorithm to handle both linearly separable and non-linear problems. Popular kernels such as the polynomial kernel, radial basis function (RBF), and sigmoid kernel map the data into higher-dimensional spaces, allowing the model to uncover complex decision boundaries. Once the optimal hyperplane is determined, new data points can be efficiently classified according to which side of the boundary they fall on.

SVMs are widely recognized for their robustness in dealing with high-dimensional datasets, where many other algorithms struggle. They are commonly applied in domains such as text classification, image recognition, bioinformatics, and financial forecasting. In addition to classification, SVMs can also be adapted for regression tasks (Support Vector Regression, SVR), demonstrating their versatility as a machine learning method. Their balance of theoretical elegance and practical effectiveness has ensured their long-standing role in the field of predictive modeling.

2.2.3 Decision tree

A decision tree is a widely used recursive partitioning structure for decision support, representing decisions and their possible outcomes in a tree-like form. Each internal node corresponds to a test on an attribute, each branch denotes the result of the test, and each leaf node represents an outcome or decision. Such a structure allows complex decision-making processes to be expressed in a simple and interpretable manner. For example, outcomes may include measures of utility, estimates of resource costs, or the likelihood of different chance events.

From a computational perspective, a decision tree can also be regarded as an algorithm that relies on a series of conditional control statements. Each condition recursively divides the problem space, ultimately leading to a decision path from the root to a leaf. This recursive nature not only makes the algorithm efficient but also ensures that the reasoning process is transparent and easy to follow.

Decision trees have found extensive applications across domains such as finance, healthcare, and engineering. They are especially valued in machine learning, where algorithms like CART, ID3, and C4.5 automatically generate decision trees from data. Their ability to handle both categorical and numerical variables, along with their interpretability, makes them a fundamental tool for predictive modeling, classification, and risk analysis.

2.2.4 AdaBoost

AdaBoost, short for Adaptive Boosting, is a statistical classification meta-algorithm first introduced by Yoav Freund and Robert Schapire in 1995. Their pioneering work in developing this ensemble method earned them the prestigious Gödel Prize in 2003, recognizing its profound impact on the field of machine learning. The core idea behind AdaBoost is to combine multiple weak learners, typically simple models such as decision stumps, into a strong classifier that achieves significantly improved accuracy.

The algorithm works by iteratively training a sequence of weak learners on reweighted versions of the dataset. After each round, misclassified samples are assigned higher weights, forcing subsequent learners to focus more on difficult instances. The final model is constructed as a weighted majority vote of all weak learners, which balances their contributions according to performance. This adaptive nature makes AdaBoost highly effective in reducing both bias and variance.

One of AdaBoost's strengths lies in its flexibility: it can be integrated with a wide range of base algorithms, including decision trees, linear classifiers, and neural networks, to enhance their predictive power. It has been successfully applied in tasks such as text classification, face detection, and bioinformatics. Despite its sensitivity to noisy data and outliers, AdaBoost remains one of the foundational ensemble methods, illustrating the power of combining simple models to achieve state-of-the-art performance.

2.2.5 Random Forest

An ensemble learning technique for classification, regression, and other problems, random forests, also known as random decision forests, generates a large number of decision trees throughout training. The class chosen by the majority of trees is the random forest's output for classification problems.

2.2.6 Naïve Bayes

Naïve Bayes, sometimes referred to as the “simple” Bayes classifier, is a family of probabilistic classifiers that relies on Bayes' theorem with the assumption of conditional independence among features. Despite the term “naïve,” which highlights the unrealistic assumption that all attributes contribute independently to the outcome, the method has proven to be remarkably effective in practice.

The core idea of Naïve Bayes is straightforward: given a set of features, the classifier estimates the posterior probability of each class by combining prior probabilities with the likelihood of observing the features under each class. The independence assumption greatly simplifies the computation, reducing what could otherwise be an exponentially complex calculation into a product of individual probabilities. This efficiency allows Naïve Bayes to perform well even on large, high-dimensional datasets.

2.2.7 Neural Network

The structure and operations of biological brain networks serve as the inspiration for the computer model known as a neural network (also known as an artificial neural network, neural net, or simply ANN or NN).

Neural networks (sometimes called artificial neural networks, neural nets, or just ANN or NN) are computer models inspired by the structure and functions of organic brain networks.

2.3. Evaluation metrics

Predictive performance is commonly evaluated using the F-score, or F-measure. It is defined as the ratio of correctly identified positive cases to all actual positive cases. Precision, in contrast, is calculated as the proportion of correctly identified positives out of all instances predicted as positive, including false positives.

3. Results and Discussion

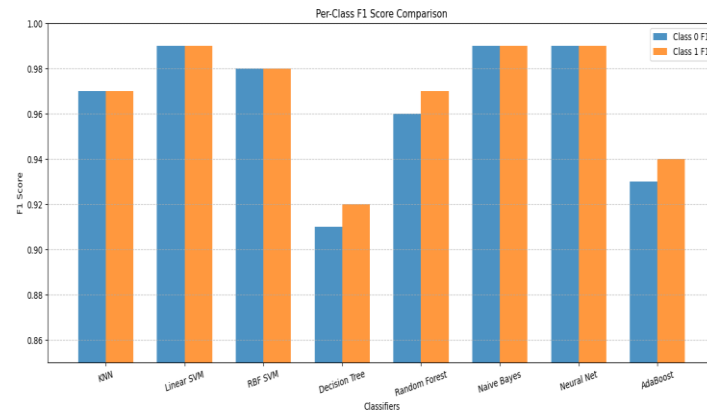


Fig 1. F1-score of different models (Picture credit: Original)

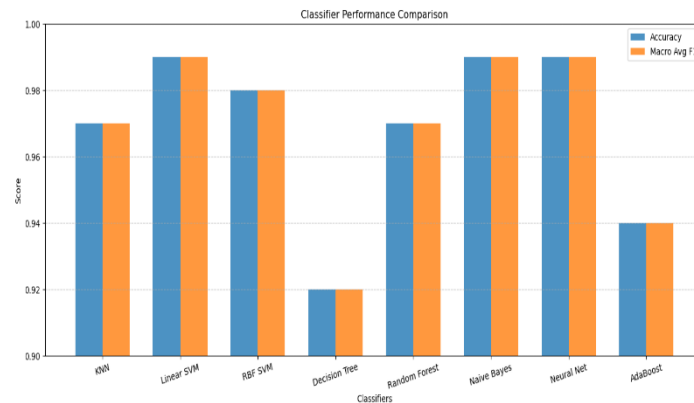


Fig 2. Accuracy of different models (Picture credit: Original)

Fig 1, Fig 2 and Table 1 provide F1-score and accuracy. Nowadays, spam email can make them seem like a common email, so this study can continue enhance the ability of solve complex problem in email.

Table 1. Model performance comparison

Model	KNN	SVM	Decision tree	Random Forest	Naive Bayes	Neural Net	AdaBoost
Accuracy	0.97	0.99	0.92	0.97	0.99	0.99	0.94
Macro Avg	0.97	0.99	0.92	0.97	0.99	0.99	0.94
Weighted Avg	0.97	0.99	0.92	0.97	0.99	0.99	0.94
Position	2	1	4	2	1	1	3

This study ranks the models and identifies the top three performers along with their advantages. Linear SVM stands out for its high-dimensional efficiency, ability to achieve a global optimal solution, and strong generalization capacity. Naïve Bayes is noted for its computational efficiency, suitability for small-sample scenarios, and consistently strong performance. Neural networks demonstrate the ability to automatically extract high-order feature combinations, model arbitrarily complex functions with an accuracy of 99% in this case, and eliminate the need for manual feature engineering.

4. Conclusion

This study compared several AI-based models for spam email detection and demonstrated that most of them achieved consistently high accuracy on the chosen dataset. Among the tested algorithms, models such as SVM, Naïve Bayes, and neural networks showed particularly strong performance, highlighting their suitability for real-world applications. The results confirm that AI can effectively address the challenges posed by unsolicited emails and significantly reduce risks for users. Future work may involve testing larger and more diverse datasets, integrating deep learning techniques, and exploring hybrid models to further improve robustness and generalization.

References

- [1] Dada EG, Bassi JS, Chiroma H, et al. Machine learning for email spam filtering: review, approaches and open research problems. *Heliyon*. 2019;5(6): e01802.
- [2] Karim A, Azam S, Shanmugam B, Kannoorpatti K, Alazab M. A comprehensive survey for intelligent spam email detection. *IEEE Access*. 2019;7:168261-95.
- [3] Raihen MN, Rana S, Akter S, Kadir MA. Efficient email spam detection using machine learning techniques: a comparative analysis of classification models. *Int J Intell Comput Inf Sci*. 2024;24(4):1-15.
- [4] Brooks RJ, Tobias AM. Choosing the best model: level of detail, complexity, and model performance. *Math Comput Model*. 1996;24(4):1-14.
- [5] Kaggle. 20-newsgroups [dataset]. 2017. Available from: <https://www.kaggle.com/datasets/crawford/20-newsgroups>