Efficient Spam Classification Using Machine Learning Methods

Zhenghao Miao*

Department of Economics, School of Arts & Sciences, University of Oregon, Eugene, United States

* Corresponding Author Email: zmiao6@uoregon.edu

Abstract. With the development of the times, we receive more and more information every day, and email is an important carrier of receiving information in our daily life. In this context, some spam came out, including some advertisements and false information. Therefore, it is necessary to classify spam and receive it selectively. Machine learning algorithms can classify and identify spam with high efficiency, and in this paper, we compare six models and achieve a top result of 98.7%. Spam classification can filter out unwanted emails at the receiving end, effectively saving people's time and preventing malicious users from stealing personal information.

Keywords: Spam Classification, Machine Learning Model, Data Classification.

1. Introduction

The number of Internet users has dramatically grown as a result of the Internet's quick global expansion, and it has taken on increasing significance in people's daily lives. Because email is such a quick and convenient means of communication, it has become indispensable to people's lives and careers as a result of the Internet's ongoing expansion. In addition, a significant amount of spam emails is constantly flooding our mailboxes, which seriously affects the normal email exchanges. According to a piece of data from the "Investigation Report on China's Anti-Spam Status in the Fourth Quarter of 2021" released by the Internet Society of China, Chinese netizens receive an average of 11.5 spam emails per week, accounting for 22.7%. Although slightly down compared to the same period in previous years, the work of anti-spam still has a long way to go.

Generally speaking, spam is an email that contains a lot of junk information, mostly advertising, publicity information, illegal information, etc. Its basic characteristics can be summarized as "unsolicited". But regarding spam, there is no official unified definition in the world at present. The Internet Society of China defines spam as:

1) Promotional emails, such as adverts, publications online, and other forms of marketing correspondence that the recipient has not specifically requested or authorized to receive;

2) Emails that the recipient cannot refuse;

3) E-mails that hide the sender's identity, address, title and other information;

4) Emails with false sources, senders, routes, etc.

In addition to consuming precious time and wasting network resources, spam directly undermines Internet security and results in significant financial losses. As spam becomes a more serious issue, more and more technologies are being used to combat it. Currently, there are two main categories of anti-spam techniques: system control techniques and mail content-based screening technologies. Such techniques include rule-based and statistics-based methods. Using rules to classify spam is simply to set a series of rules, such as keywords, information sources, etc, once the mail meets the rules, it will be judged as spam. This kind of classification is simple and easy, low cost and high efficiency, but it has a high misjudgment rate and poor adaptability, and it is easy to judge normal emails as spam. Commonly used methods are Boosting [1], Decision Tree [2] and so on. The statistical-based method classifies emails through a large number of normal emails and spam training sample sets. This classification method has high flexibility and strong timeliness, but has high computational complexity and general applicability to different users. Commonly used methods are SVM, Naive Bayes, etc. Due to the fact that there are many drawbacks in the current method of filtering spam using the system control method on the server side, this paper will focus on the

introduction of content-based spam filtering technology, mainly some machine learning methods. Finally, through experimental results, we believe that Naive Bayes is the best algorithm in this field.

2. Method

2.1. Data description

There are 4458 pieces of data in the overall data set, which will be split into a train set and a validation set at an eight-to-two ratio. Through analysis, it is found that the number of pams is 3866, accounting for the majority of the data set, and unbalanced sample sampling can be considered for training.

The data has three columns: ID, Label (pam, spam), Email:

```
ID,Label,Email
0,ham,"I don't have anybody's number, I still haven't thought up a tactful way to ask alex"
1,spam,"Congrats! 2 mobile 3G Videophones R yours. call 09063458130 now! videochat wid your m
2,ham,"She is our sister.. She belongs 2 our family.. She is d hope of tomorrow.. Pray 4 her,
3,ham,Ya very nice. . .be ready on thursday
```

Figure 1: UCI Spam Classification Data

2.2. Text preprocessing

The process of choosing one or more category labels from a list of pre-established categories to apply to test documents is known as text categorization. Simply put, it is the process of classifying unclassified text into its correct category. This process includes text preprocessing, classifier selection and training, and evaluation and feedback of classification results. Classification of spam can simply be viewed as a binary text classification problem, but it is different from the ordinary binary classification problem of classifying a given text into two known classes. Usually in the spam corpus, the proportion of spam can account for 80%-90%, so its feature selection is aimed at the feature selection problem of an unbalanced corpus. Not only that, the classification of spam has obvious characteristics, and the era, geographical location, time change, etc. may have a great impact on it.

The object processed by the text classification task is the text with natural language representation, which belongs to unstructured data. Therefore, before using the corpus for text classification, the text is structured. Text preprocessing generally includes lexical analysis, stop word removal, stemming, word segmentation, etc.

2.3. Feature selection

The document is combined with some features after preprocessing, and each feature space is assigned one dimension based on its word frequency, inverse document frequency, etc. The document in this instance has to have its dimensions decreased because of its extremely high dimensionality. The two kinds of frequently used feature dimensionality reduction techniques are feature extraction and feature selection. In order to achieve the goal of dimensionality reduction, feature extraction refers to the application of transformation. This method is not as popular as feature selection strategies because of the significant computing complexity. Using a feature selection function to rate each feature in a document collection and choosing the features with higher scores to represent documents is known as feature selection.

2.4. Our models

Bayesian. Bayesian algorithm is a widely used classification algorithm. Its principle is to first calculate the prior probability of a certain phenomenon, and then use the Bayesian formula [3] to calculate its posterior probability. That is, first calculate the probability that the text m belongs to each category, and then sort these probabilities from large to small, and select the category with the highest probability as the category of m. The most widely used Bayesian classification algorithm is

the naive Bayesian classifier, in which the text m consists of a series of independent features tiYi=1,2, 3, nY, such that Computing P(cj|m) is transformed into computing P(cj|ti).

SVM. Support vector machine [4] is a pattern recognition method based on statistical machine learning theory, which is widely used in pattern recognition, text classification and bioinformatics. According to its basic tenets, the classification boundary is established by first linearly separating the low-dimensional points in the low-dimensional space by mapping them to the high-dimensional space. The low-dimensional space here indicates that the feature dimension is lower for emails, and the high-dimensional space indicates that the feature dimension is higher. In low-dimensional space, the models we train are mostly linear models, which may misclassify some outliers; if it is a kernel function in high-dimensional space, we can classify these outliers more accurately. The processing complexity is increased and the process of calculating the mapping is no longer necessary thanks to the kernel function, which makes it possible to conduct all operations in the high-dimensional space in the low-dimensional space.

K nearest neighbors. The K-adjacent neighbor algorithm [5] is a classification algorithm that is widely used and has a wide range of applications. Its principle is easy to understand: compare the distance between text m and all other texts, and take the first k texts with the shortest distance from text m. This distance indicates how similar the two samples are. This is a very simple distance calculation method, and methods such as cosine similarity can also be used to compare the "distance" between two samples, that is, whether the two samples are similar. If the distance between the two samples is short, then we have reason to think that the two samples belong to the same category. As a similar text of text m, the category of text m is determined according to the categories of the k texts. The KNN algorithm does not have a training process. Although it often achieves very good results, it has a high computational complexity. This algorithm is not suitable for the field of spam classification that requires high computational speed.

3. Experiments

3.1. Comparison results:

Various machine learning algorithms were used in this experiment, including Naive Bayes, Logistic Regression [6], Support Vector Machine, KNN, Decision Tree [7], Random Forest [8], etc. The Naive Bayes method has the maximum accuracy, as shown by the results.

Table 1: Results of different algorithms and comparisons

	accuracy	precision	recall	f1-score
Naive Bayes	0.987	0.962	0.950	0.953
LR	0.979	0.988	0.859	0.919
SVM	0.982	1.0	0.864	0.928
KNN	0.910	1.0	0.330	0.496
Decision-Tree	0.969	0.899	0.870	0.884
Random-Forest	0.982	1.0	0.865	0.927

Accuracy-score is the ratio of correctly predicted samples to all anticipated samples, whether the anticipated samples are positive or negative. By dividing the number of correctly predicted positive samples by the total number of expected positive samples, the precision-score calculates the proportion of all anticipated positive samples that are really true positive samples. It is evident that although precision only takes into consideration the percentage of the sample that is predicted to be positive, accuracy incorporates all samples. By comparing positively predicted samples to r all really positive samples, recall-score, also known as recall rate and recall rate, determines how many positive samples can be reliably recognized from those samples. The harmonic average of precision and recall is the same as the F1-score. The F1-score will drop if either recall or precision declines, and vice versa.

$$\text{accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \quad (1)$$

$$\text{precision} = \frac{TP}{TP+FP} \quad (2)$$

$$\text{recall} = \frac{TP}{TP+FN} \quad (3)$$

$$\text{F1-score} = \frac{2}{1/precision+1/recall} \quad (4)$$

3.2. Discussion:

The experimental outcomes demonstrate that SVM [9], KNN and Random Forest perform the best in discriminating non-spam emails, but Decision Tree performs the worst in this regard; Naive Bayes performs the best in terms of recall, and KNN performs the worst. , mainly because KNN will misclassify some spam as non-spam; similarly, F1-score[10] is the harmonic average of recall-score and precision-score, so KNN's F1-score is also very low; in terms of overall accuracy , Naive-Bayes performs the best, KNN performs the worst, and SVM is also close to the optimal value, so we have reason to believe that SVM is a suitable algorithm in the field of spam classification.

4. Conclusion

The issue of spam classification is currently a focus of research for a number of mail service companies. The fundamental concepts and procedures of spam categorization are introduced, along with six popular machine learning algorithms (Naive Bayes, Logistic Regression, Support Vector Machine, KNN, Decision Tree, Random Forest)—and experimental comparisons are made. Contrary to typical text classification, spam classification is a lengthy and dynamic process. Spammers continue to produce more spam that confuses classifiers more readily as we develop feature selection methods [11] and classifier accuracy. Therefore, in addition to increasing the algorithm's accuracy rate, we need to create a perfect spam categorization system [10] and perform an excellent job of preventing and controlling spam at its source, during transmission, and at its destination. In future work, we can use deep learning methods to further study spam classification, collect a large amount of data and improve text processing, and finally input it into the neural network, the author believes there will be better results.

References

[1] Ali A B M S, Xiang Y. Spam classification using adaptive boosting algorithm [C]//6th IEEE/ACIS International Conference on Computer and Information Science (ICIS 2007). IEEE, 2007: 972-976.

[2] Subasi A, Alzahrani S, Aljuhani A, et al. Comparison of decision tree algorithms for spam e-mail filtering[C]//2018 1st International Conference on Computer Applications & Information Security (ICCAIS). IEEE, 2018: 1-5.

[3] Seewald A K. An evaluation of Naive Bayes variants in content-based learning for spam filtering [C]// IOS Press. IOS Press, 2007:497-524.

[4] Tseng C Y, Chen M S. Incremental SVM Model for Spam Detection on Dynamic Email Social Networks [C]// 2009 International Conference on Computational Science and Engineering. IEEE Computer Society, 2009.

[5] Zhang J. Application of Improved KNN Algorithm in Spam E-mail Filtering [J]. New Technology of Library and Information Service, 2007.

[6] Chang M W, Yih W T, Meek C. Partitioned logistic regression for spam filtering [C]// Acm Sigkdd International Conference on Knowledge Discovery & Data Mining. ACM, 2008.

[7] Bao L Q, Chai S H. The Application of Decision Tree in Spam Filtering [J]. Journal of Lanzhou Polytechnic College.

[8] Kumar M. Effective Spam Filtering using Random Forest Machine Learning Algorithm.

[9] Tsang, I. W., Kwok, J. T., Cheung, P. M., & Cristianini, N. (2005). Core vector machines: Fast SVM training on very large data sets. Journal of Machine Learning Research, 6(4).

[10] Lipton, Z. C., Elkan, C., & Narayanaswamy, B. (2014). Thresholding classifiers to maximize F1 score. arXiv preprint arXiv:1402.1892.

[11] Li, J., Cheng, K., Wang, S., Morstatter, F., Trevino, R. P., Tang, J., & Liu, H. (2017). Feature selection: A data perspective. ACM computing surveys (CSUR), 50(6), 1-45.

[12] Renuka, D. K., Hamsapriya, T., Chakkaravarthi, M. R., & Surya, P. L. (2011, July). Spam classification based on supervised learning using machine learning techniques. In 2011 International Conference on Process Automation, Control and Computing (pp. 1-7). IEEE.