FAST EMAIL SPAM FILTERING METHODS

Aditya Bhute, Nagraj Aajure, Shubham Dhanorkar, Prof.Kapil Wagh

Information Technology, Pimpri Chichwad Trust's Nutan Maharashtra Institute of Engineering and Technology, India

ABSTRACT

The paper elaborates on how text analysis influences classification—a key partof the spam-filtering process. The authors propose a multistage meta-algorithm for checking classifier performance. As a result, the algorithm allows for the fast selection of the best-performing classifiers as well as for the analysis of higher-dimensionality data. The last aspectis especially important when analyzing large datasets. The approach of cross-validation between different datasets for supervised learning is applied in the meta-algorithm. Threemachine-learning methods allowing a user to classify e-mails as desirable (ham) or potentially harmful (spam) messages were compared in the paper to illustrate the operation of the meta-algorithm. The used methods are simple, but as the results showed, they arepowerful enough. We use the following classifiers: k-nearest neighbours(k-NNs), support vector machines (SVM), and the naïve Bayes classifier (NB). The conducted research gave us the conclusion that multinomial naïve Bayes classifier can be an excellent weapon in the fight against the constantly increasing amount of spam messages. It was also confirmed that the proposed solution gives very accurate results.

Keywords: *classifiers; e-mail; sms; ham; machine learning; spam*

INTRODUCTION:

The spam problem is an ongoing issue: in 2018 14.5 billion spam e-mails were sent per day. According to the Internet Security Threat Report released in 2019 by Symantec, spam levels for their customers increased in 2018. What draws the attention is that small enterprises were attacked more often than large companies,and e-mail malware reached stable levels. Therefore, there is a need to tailor even simple tools for detection and filtering of spam in all organizations.

Identification of the best-performing machine learning-based classifiers and selection of the one with the leading parameters. The proposed solution solves theproblem of fast recognition of the most interesting parameters. This allows for quick analysis of data of higher dimensionality. This is especially important if large datasetsare to be analyzed and we want to assure the proper scalability of our system.

PROBLEM STATEMENT:

That this is any "attempt to abuse, or manipulate, a techno-social system by producing and injecting unsolicited and/or undesired content aimed at steering the behavior of humans or the system itself, at the direct or indirect, immediate or long-term advantage ofthe spammer(s)". Here, we focus on so-called junk e-mails. These are unwanted messagessent at large scale by e-mail. The term spam refers to the undesired (or even harmful) e-mails,while ham is used to indicate the valid and important messages desired by the recipient. Additionally, we assume the scenario where junk e-mails are sent by botnets

and they arenot aimed at specific users (contrary to, e.g., spear phishing).

METHODS AND EQUATIONS:

E-mail spam filtering is a compound task, and in general we follow the methods elaboratedbefore, where is the main source of inspiration for us. The main goal of this paper is to exploreone of its key areas, i.e., machine-learning-based classification, to help with the initial decision if a given e-mail message is indeed spam or ham. The element that enables thisresearch is a dataset selected as a pool for training. The dataset is a collection of real e-mail examples. Access to a useful dataset is not a trivial issue, since typically in the academicalworld it is not possible to obtain e-mails for scientific research. Additionally, it is necessary to gain access to the database where the messages are already labeled as spam or ham.

Here, we propose a multistage meta-algorithm that allows us to select the best hyperparameters for various classification algorithms and then compare their performanceto decide on which one to use. The meta-algorithm is presented in Figure 1. Please note that the classification algorithms shown are only used as illustration. The following stages of themeta-algorithm are as follows:

- 1. Selection of a database.
- 2. Text analysis.
- 3. Spam detection: cross-validation on different datasets.
- 4. Final selection.

Text preprocessing plays a crucial role in spam filtering [24,37]. For any spam detection model to be effective, the content of the e-mails should be normalized and represented asfeature vectors. The starting point is the tokenization of the raw text data. Then there areseveral steps shown in Figure2to obtain the data in the form that is ready to be analyzed by the model.

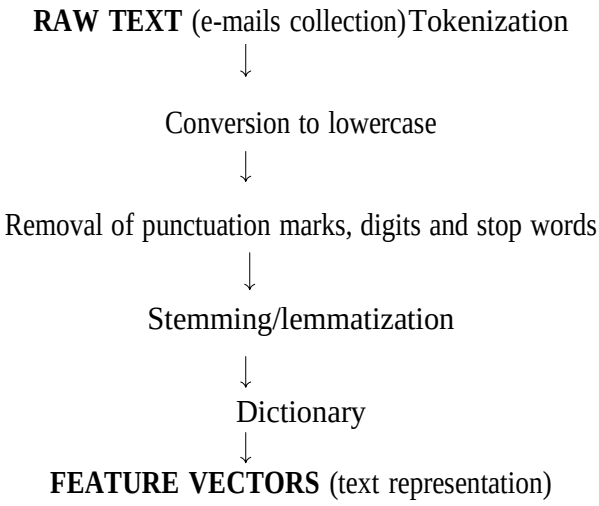

Figure 2. Text preprocessing steps.

Tokenization technique allows us to split the content of the e-mails into basic processing units that are called tokens or features. Given that the paper deals with text data, the tokens are simply separate words. For instance, the tokenized sentence "Subject: Christmas tree farm pictures" is a collection of strings: "Subject", ":", "Christmas", "tree", "farm" and "pictures". The next step involves converting all tokens to lowercase. As a result of this simpleoperation, the number of words taken into account is significantly reduced. Instead of treating "Example", "example" and "EXAMPLE" as three different words, after converting them to lowercase, we make sure that the program will count them as one ("example"). Punctuation marks, digits, and stop words are all common in both spam and ham e-mailsand do not add any value to text analysis. Since we implement our solution in Python, werefer to tools related to this programming language. There are several libraries and functions that may be applied to eliminate the mentioned language elements not essential from thespam detection viewpoint. Below is the list of functionalities chosen by us.

Python method string.isalpha() checks whether the characters in the string are alphabetic or not. If the character is a digit, the method returns False. The method string.punctuation() allows removal of common punctuation marks, such as commas, periods, semicolons, etc. Natural Language Toolkit (NLTK) offers a module containing a list of stop words that are the most common words in a language. The examples of stop words are short words, for example: "the", "is", "at", "which", or "on" [38]. The universal list of stop words does not exist,any set can be adopted depending on the purpose. Next, stemming reduces the morphological variants of the word to its base (stem). The algorithms enabling that operation are often called stemmers. In Python, that may be implemented with the use of NLTK [39]. ForEnglish language, there exist two stemmers: PorterStemmer and LancasterStemmer. For the purpose of this paper, the PorterStemmer (PS) was chosen and tested with the designed models because of its simplicity and the speed of its operation. PS is dated to 1979 and often generates stems that are not authentic Text preprocessing plays a crucial role in spam filtering [24,37]. For any spam detection model to be effective, the content of the e-mailsshould be normalized and represented as feature vectors. The starting point is the tokenization of the raw text data. Then there are several steps shown in Figure2to obtain the data in the form that is ready to be analyzed by the model. English words. It results from the fact that it is based on suffix stripping (examples shown in Table1). Instead of considering linguistics to build the stem, it applies a set of algorithmic rules that decide if it is reasonable toremove the suffix or not.

Table 1. Examples of stemming with PS.

Word Before	**Word After**
Cats	Cat
Trouble	Troubl
Troubling	Troubl
Troubled	Troubl

Other option, known as lemmatization, is a more complex approach to searching a word'sstem. In this

case, the root word is referred to as a lemma. First, the algorithm identifies the part of the speech of a word; and then, based on this information, it applies appropriate normalization. As in the stemming case, lemmatization mechanisms are also provided byNLTK [39]. WordNet Lemmatizer (WNL) generates lemmas by searching for them in the WordNet Database. Examples are shown in Table2. In the research reported here, text preprocessing was supported by the most basic lemmatization version in specific test cases. However, the method works most efficiently when one defines the context by assigning thevalue to pos parameter (for instance by giving it the value v—verb). Testing with the posvalue defined is outside of the scope of this paper, but its usefulness may be noticed afterthe analysis of the impact the pos = v has on the verbs shown in Table2.

Table 2. Examples of lemmatization with WNL.

Value pos Undefined

Word Before	**Word After**
He	He
Was	Was
Has	Has
Playing	Playing

pos = v

Word Before	**Word After**
He	He
Was	Be
Has	Have
Playing	Play

One may ask which one is better: stemming or lemmatization? The answer is that it depends on the program and the requirements that one is working with. If speed is a priority, then itis more beneficial to use stemming. When language is crucial for the application's purpose,lemmatizing should be a choice as it is more precise.

In e-mail spam filtering, the goal of building the dictionary structure (key-value with uniquekeys) consists of assessing the word's weight and importance given all available text documents. First, word occurrences are calculated. In the case of the application presentedhere, words are limited to strings of the length between 3 and 20 characters. Single lettersand extremely short/long strings do not add value to the paper (they are common for bothham and spam).

First, we create two separate dictionaries (spamWords and hamWords). The function responsible for the dictionary generation returns the number n (defined during the tests) of the most common words for each of them. Next, another function builds dictionaries which include common words (subtractFromSpam, subtractFromHam). Based on these structures,three others are defined:

1.spamDictionary = spamWords-subtractFromSpam 2.hamDictionary = hamWords-subtractFromHam 3.finalDictionary = spamDictionary + hamDictionary.

According to the informal research carried out by Dave C. Trudgian [40], the unbalanced distribution of spam and ham most common words significantly affects the models' accuracy. The results were improved when the final dictionary included more spam's most common words than ham's most common words. Table3presents the ratios implemented inthe application described in this paper.

Table 3. Implemented most common words ratios (spam:ham).

No.	Spam	Ham	Total
1	150	50	200
2	900	600	1500
3	2000	1000	3000

OBJECTIVES:

- The increasing number of spam e-mails has created a strong need to develop morereliable and efficient anti-spam filters, including ones based on machine-learning tools.
- They are efficient, since they only require the preparation of a set of training samples, i.e., pre-classified e-mails. In recent years, various machine-learning methods have been successfully used to effectively detect and filter unwanted messages.
- The following classification methods are most commonly used for spam filtering: Support Vector Machine(SVM), Naïve Bayes classifier (NB), k-Nearest Neighbours (k-NN), Artificial Neutral
- Network (ANN), Decision Tree (DT), Random Forest (RF), Logistic Regression (LR).
- The values are given at the end of the numerical study in separate table.
- The applicability of using different machine-learning methods to recognize spam e-mails was analyzed in. The SpamAssassin dataset, which contains 6000 e-mails with the spamrate 37.04% used in all experiments.
- Sharma and Arora in analyzed Bayes Net(BN), Logic Boost (LB), RT, JRip (JR), J48- based DTs, Multilayer Perceptron (MP), Kstar (KS),RF, and Random Committee (RC) machine-learning algorithms. The dataset with 4601instances and 55 spam base attributes downloaded from UCl Machine-Learning Repository were used in the performed research.
- Harisinghaney et al. applied the following three different algorithms: k-NN, NB, and DBSCAN-based clustering. The performance for the four metrics accuracy, precision, sensitivity, and specificity were calculated and compared.

LITERATURE SURVEY:

Title: A Method for Fast Selection of Machine-Learning Classifiers for Spam Filtering.

Author name: Sylwia Rapacz , Piotr Chołda and Marek Natkaniec

Description: The paper[1] addresses development of system that deals with spamfiltering methods on surveys and results of various tests conducted by them on readymade databases.

Title: Detecting Spam Email With Machine Learning Optimized With Bio-Inspired Metaheuristic Algorithms.

Author name: SIMRAN GIBSON 1, BIJU ISSAC 1, (Senior Member, IEEE),LI ZHANG 1, (Senior Member, IEEE), AND SEIBU MARY JACOB 2, (Member, IEEE)

Description: The paper[2] successfully implemented models combined with bio-inspired algorithms. The spam email corpus used within the project were both numerical as well as alphabetical. Approximately 50,000 emails were tested with the proposed models. The numerical corpuses (PU), had restrictions in terms of feature extraction as the words were replaced by numbers. But the alphabetical corpuses performed better in terms of extractionof the features and predicting the outcome.

Title: SMS Spam Filtering Using Supervised Machine Learning Algorithms.

Author name: Pavas Navaney, Ajay Rana, & Gaurav Dubey.

Description: This paper[3] is all about SMS spam filtering using Machine Learning Techniques. Various methods of spam filtering algorithms are discussed.

- IEEE 2021 papers/Research gate : A Method for Fast Selection of Machine-Learning Classifiers for Spam Filtering electronics-10-02083-v2.
- Artificial Intelligence applications in supply chain 2021
- Symantec. Internet Security Threat Report. 2019. Available online: https://www.symantec.com/content/dam/symantec/docs/reports/istr-24-2019- en.pdf
- Multi-lane Capsule Covid-19 Detection.
- IEEE 2021 paper Email spam filtering using AI
- IEEE 2020 -2021 paper Email spam filtering using ML
- Highly Isolated Self-Multiplying 5G Antenna for IOT Applications
- Analysis of Attacks on components of IOT systems and Cybersecurity Technologies.

- Formation of International Ethical Digital Environment with Smart Artificial Intelligence.
- SMS filtering using ML (2020) rereleased-2021
- Ferrara, E. The History of Digital Spam. Communication. ACM 2019
- Exploratory Analysis on Increasing Attacks on Power Grid
- Exploratory Analysis based on remote health care
- Exploratory Analysis on Attacks on IOT devices
- Data Analysis on Spam classifier
- Wearables and the IOT
- Digit Recognition From Movements and Security

APPLICATION:

- Improved Spam filtering methods in E-mail.
- Next level security against phishes, virus bound email.
- Future system with common spam filtering for SMS and Emails.
- Safe systems.
- Virtual Machines
- Virus Detection
- Improved UI

FUTURE SCOPE:

New improvised technology that does spam detection in Messages and Emails which alsoautomatically level up the field according to threats.

REFERENCES:

- Internet
- IEEE
- ResearchGate
- Bauer, E. 15 Outrageous Email Spam Statistics that Still Ring True in 2018. Available online: https://www.propellercrm.com/blog/email-spam-statistics
- Dada, E.G.; Bassi, J.S.; Chiroma, H.; Adetunmbi, A.O.; Ajibuwa, O.E. Machine Learning for Email Spam Filtering: Review, Approaches and Open Research Problems. Heliyon2019, 5, e01802. [CrossRef] [PubMed]
- Awad,W.A.; ELseuofi, S.M. Machine Learning Methods for Spam E-Mail Classification. Int. J. Comput. Sci. Inf. Technol. 2011, 3,173–184. [CrossRef]