International Journal for Modern Trends in Science and Technology, 9(07): 18-22, 2023
Copyright © 2023International Journal for Modern Trends in Science and Technology
ISSN: 2455-3778 online
DOI: https://doi.org/10.46501/IJMTST0907004
Available online at: http://www.ijmtst.com/vol9issue07.html

Categorization of Phishing Websites using an Extremely Intelligent Machine Learning Algorithm

Sravani Vankayalapati[1] | Dr.Suneetha Davuluri[2]

[1]PG Student, Dept of CSE, NRI Institute of Technology, Vijayawada, A.P.
[2]Professor&CSE HOD, Dept of CSE, NRI Institute of Technology, Vijayawada, A.P.
Email: vankayalapatisravani12@gmail.com1, hod.csenriit@gmail.com2

To Cite this Article
Sravani Vankayalapati and Dr.Suneetha Davuluri. Categorization of Phishing Websites using an Extremely Intelligent Machine Learning Algorithm. International Journal for Modern Trends in Science and Technology 2023, 9(07), pp. 18-22. https://doi.org/10.46501/IJMTST0907004

Article Info
Received: 30 May 2023; Accepted: 28 June 2023; Published: 01 July 2023.

ABSTRACT

As cyberattacks go, phishing is up there with the worst of them. The goal of these assaults is to get financial data utilized in business and personal activities. The content and settings of a web browser might provide indicators about the legitimacy of a website. This research endeavours to classify 30 characteristics, such as Phishing Websites Data from the UC Irvine Machine Learning Repository, using an Extreme Learning Machine (ELM). When compared to another machine learning technique called Naive Bayes (NB), ELM was shown to have a higher accuracy of 85.73%when evaluating the outcomes.

Keywords: Cyber attacks, Phishing attacks, Naïve Bayes, Machine Learning

1. INTRODUCTION

Phishing attacks and other types of identity theft-based frauds are growing in popularity among hacker groups due to the increasing use of the Internet for online banking and commerce. More than 50 million phishing emails were sent in 2004. The harm they caused to financial institutions was $10 billion. These days, most phishing attempts consist of a three-stage procedure. First, the phishers use social engineering techniques, malicious websites, and online discussion boards to send emails to their targets. Massive amounts of phishing emails posing as from legitimate financial institutions are sent from hidden servers or hacked computers. The websites linked to in these emails seem quite similar to the real thing. Forms asking for sensitive information including credit card numbers, social security numbers, birthdates, and more can be seen on the bogus website. Although phishing emails can be fought using spam filtering methods already in place, these safeguards have limited reach. There are a number of methods that may be used to avoid detection by statistical and rule-based spam filters. Despite these safeguards, phishing emails continue to pose a significant concern since they are not specifically designed to identify such messages. While spam indirectly affects its targets by reducing available bandwidth, phishing attempts directly influence their victims by costing them a significant amount of money. Phishing assaults are a major issue because of the ease with which fraudsters may create convincing socially engineered communications by exploiting technological flaws (for example, by utilising a seemingly valid but really faked domain name). For mitigation efforts to be really effective, they must target both the technological and human layers. Phishing attacks are challenging to prevent because they rely on fooling people (the end users of the targeted system). Even after being exposed to the most effective user awareness program, consumers still missed 29% of phishing assaults. However, software phishing detection systems are tested using actual phishing attempts, so it's impossible to determine how well they'd do against more nuanced attacks. Due to these shortcomings, several businesses, including industry leaders in information security, have come dangerously close to suffering security breaches that might have been prevented. The unique plugin of the Google Chrome web browser is based on Blacklisting and semantic analysis approaches that will be effectively integrated to efficiently detect and avoid the phishing assault; this is one of the primary contributions we emphasise. Phishing detection checks IP addresses and whether or not the user's data is redirected. The phishing attack detection model is outlined here. The suggested methodology is geared at spotting phishing attacks by comparing red flags with a blacklist of known malicious domains. Our proposed tool suggests using just a handful of carefully chosen characteristics to identify phishing from non-phishing websites. Included in this group are universal resource locators (URLs), domain names, page layout and content, the URL bar, and the human element. In this work, we examine solely aspects related to domain names and URLs. Multiple criteria, including IP address, lengthy URL address, redirecting using the symbol "//," and URLs bearing the mail/mail-to characteristics, are used to validate domain names and URLs. Worldwide, both consumers and businesses are targets of phishing attacks. Because it crosses international boundaries, tracing the culprits is tough. The "fast-flux" technique used by the phishers also involves a huge number of proxy servers and URLs used to conceal the true address of the phishing site. At the same time, the server being utilised makes it more difficult to ban the site. Phishing attacks exploit weaknesses in computer systems caused by human error. Users are the most vulnerable part of any network since many cyberattacks utilise techniques that propagate by exploiting vulnerabilities in end users. Various groups have tried various approaches to the issue in order to find a solution. Most Google Chrome extensions in the anti-phishing category work to protect users against scams on social media and auction sites. The site's URL may also be checked as a second way. The URL is parsed into individual words, and those words are then followed as links; if a connection is made, the site is flagged as phishing. There are a number of problems with this approach, the most significant of which are the lack of adaptability and the poor quality of their output.

2. LITERATURE REVIEW

Arms race is a common metaphor for the competition between spammers and those who try to stop them. We keep coming up with new anti-spam measures, but spammers keep finding methods to circumvent them. This is seen by their efforts to fool spam filters. Spammers have attempted everything from clever HTML layouts and letter replacement to arbitrary data injection. Although such assaults may be ingenious at times, they have not proven to be very effective against the statistical nature that underpins many filtering systems. The wide range of filtering systems makes it unlikely that a single assault would work against all of them, which significantly increases the difficulty of constructing such an attack. Here, we look at the broad strategies spammers use in their attacks, along with the problems both developers and spammers have to deal with. We also show an approach that, although simple to deploy, makes more robust efforts to undermine filters' inherent statistical foundations[1].

There are a plethora of anti-phishing tools out there. However, there are fewer research comparing machine learning algorithms in phishing prediction than there are in spam prediction. Logistic Regression (LR), Classification and Regression Trees (CART), Bayesian Additive Regression Trees (BART), Support Vector Machines (SVM), Random Forests (RF), and Neural Networks (NNet) are just some of the machine learning techniques that were tested and compared for their predictive accuracy in identifying phishing emails. The research makes use of a dataset including 2889 phishing and authentic emails for analysis. The classifiers are trained and tested on the basis of 43 characteristics[2].

Improved Phishing Detection using ModelBased Features. In CEAS. Emails that seem legitimate but are really scams pose a serious risk to online commerce and communication. Criminals are actively targeting naive internet users in an effort to get sensitive information such as passwords, account numbers, and social security numbers. Since a new phishing scam is developed every minute on average, blacklist-based techniques to filtering are insufficient. We look at statistical phishing email filtering, where a classifier is trained on traits typical of known phishing emails and can then recognise new phishing emails with novel content. Adaptively trained Dynamic Markov Chains and innovative latent Class-Topic Models are proposed as a means of producing cutting-edge features for use in email. Using these attributes, classifiers may improve accuracy by two-thirds on a publicly accessible test corpus including misclassified emails. By using a newly established more expressive assessment approach, we are able to demonstrate the statistical significance of these findings. In addition, we successfully piloted our method on a real-world, non-public email corpus[3]. Cantina+: A feature-rich machine learning framework for detecting phishing web sites. ACM Transactions on Information and System Security (TISSEC), 14(2), 21. Online phishing attacks are rampant. Most phishing detection strategies rely on either blacklists of known malicious URLs curated by humans or automated analyses of a website's structure and content. However, the former is vulnerable to new phish, while the latter has a low detection rate and few useful features. (FP). To address these issues, we offer a multi-tiered anti-phishing approach that 1) uses machine learning to make the most of a large feature set and obtain a high true positive rate (TP) on new phish, and 2) uses filtering methods to keep the false positive rate (FP) to a minimum. To that end, we present CANTINA+, the most complete feature-based method to phish detection in the literature, which makes use of the HTML Document Object Model (DOM), search engines, and third-party services in conjunction with machine learning approaches. Additionally, we created two filters to aid in FP reduction and execution time optimisation. The first is a hashing-based system for detecting suspiciously similar phishing emails. The second is a filter for identifying login forms on websites, which automatically verifies as safe any page that doesn't display such a form[4]. More and more assaults are conducted every month with the intention of tricking internet users into sharing their personal information by making them assume they are dealing with a reputable organisation. Phishing is a kind of attack in which sensitive information is stolen via the use of emails that include links to malicious websites. To counter these threats, we detail a technique that, at its core, use machine learning to analyse a feature set created to draw attention to user-targeted deception in digital communication. With few tweaks, this approach may be used to identify phishing websites or the emails that lead potential victims there. We test our approach on a dataset consisting of 860 phishing emails and 6950 non-phishing emails, and find that it successfully identifies over 96% of the phishing emails while incorrectly categorising just around 0.1% of the valid emails. We wrap up with some reflections on the long-term prospects for such systems to precisely detect fraud, bearing in mind the ever-changing nature of both assaults and information[5].

3. PROPOSED METHOD

In this research, the input and output parameters for the ELM classifier are determined and then used to categorise characteristics from the phishing website database. Comparing ELM's results to those of other classifiers (SVM and NB), the former is shown to be more successful. Researchers believe the approach used here might be implemented in automated systems to great effect in the fight against phishing on the web. This research also has the greatest test performance among similar studies in the literature, with an 80.18% success rate.

Learning Algorithms

Machine learning methods Support Vector Machine (SVM) and naive Bayes form the basis of the proposed system.

Support Vector Machine (SVM)

Exceptional hyperplanes are used to divide all features of a single kind as part of the SVM categorising data. If you're using a support vector machine (SVM) technique, the optimal hyperplane is the one with the longest line connecting the classes. In order to sort information into categories, a support vector machine (SVM) looks for the exceptional hyperplane that divides the many features of knowledge into their respective groups. Support vectors are the informational features that are closest to the keeping-apart vectors.

Naïve Bayes

Naive Bayes is a probabilistic classifier that assigns an item to a group or category depending on how probable it is to belong to a certain class. Naive Bayes, as the name suggests, is an algorithm that relies only on a feature's independence from any others. The time, date, language, and location of postings are only some of the ways that we may spot fake accounts. All of these characteristics, in my view, increase the likelihood that the false profile exists, even if they rely on each other or on the existence of the other features.

$$P(A|B)=\frac{P(B|A)P(A)}{P(B)}$$

Where,

P(A|B) is Posterior probability: Probability of hypothesis A on the observed event B.
P(B|A) is Likelihood probability: Probability of the evidence given that the probability of a hypothesis is true.
P(A) is Prior Probability: Probability of hypothesis before observing the evidence.
P(B) is Marginal Probability: Probability of Evidence.

1) Steps to implement:

- Data Pre-processing step
- Fitting Naive Bayes to the Training set
- Predicting the test result
- Test accuracy of the result(Creation of Confusion matrix)
- Visualizing the test set result.

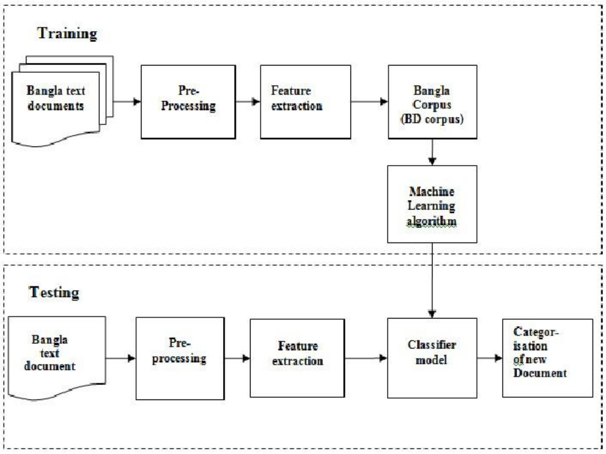
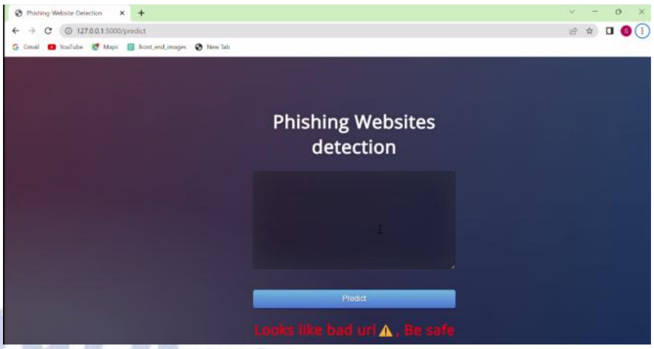


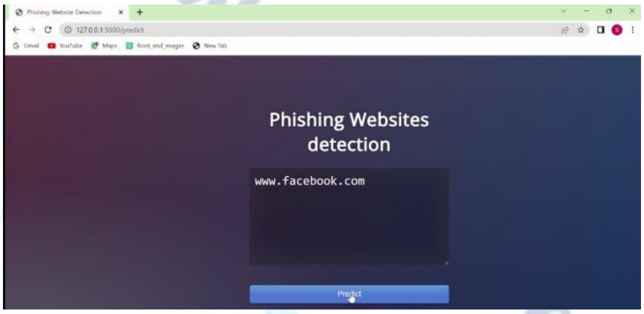
Fig1: System Architecture

4. EXPERIMENTAL RESULTS



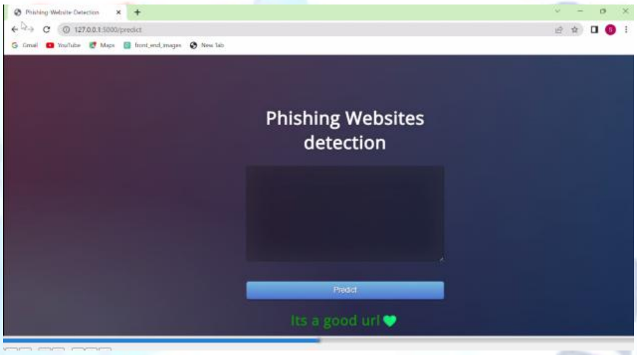
Phishing Websites detection

Looks like bad url ⚠ , Be safe



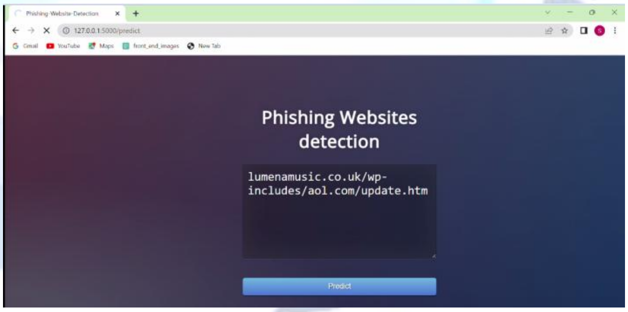
Phishing Websites detection

www.facebook.com



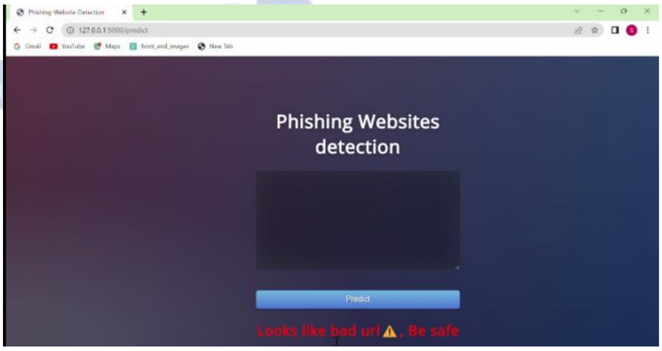
Phishing Websites detection

Its a good url 💚



Phishing Websites detection

lumenamusic.co.uk/wp-includes/aol.com/update.htm



Phishing Websites detection

Looks like bad url ⚠ , Be safe

5. CONCLUSION

In this research, we present Anti-Phishing Extension to deal with phishing materials. Three separate algorithms—"Phishing detection," "URL for IP address," and "user information redirection"—make up the suggested method. Protecting consumers against phishing attempts to address the human component is the primary topic of this article. This includes the theft of personal information from bank accounts, credit cards, social media, etc. The suggested APE method improves the speed and accuracy with which phishing assaults are identified. The suggested APE method is compatible with the add-on for Google Chrome. Our solution is shown using JavaScript-based code. The findings show that our suggested APE strategy outperforms competing methods in terms of accuracy while consuming much less CPU. We want to acquire more capabilities by identifying page content based on visual cues such as photographs and videos in the near future.

Conflict of interest statement

Authors declare that they do not have any conflict of interest.

REFERENCES

[1] K. Albrecht, N. Burri, and R. Wattenhofer. Spamato - An Extendable Spam Filter System. In 2nd Conference on Email and Anti-Spam (CEAS), Stanford University, Palo Alto, California, USA, July 2005.

[2] A. Alsaid and C. J. Mitchell. Installing fake root keys in a pc. In EuroPKI, pages 227–239, 2005.

[3] Anti-Phishing Working Group. Phishing activity trends report, Jan. 2005. http://www.antiphishing.org/reports/apwg_report_jan_2006.pdf.

[4] Apache Software Foundation. Spamassassin homepage, 2006. http://spamassassin.apache.org/.

[5] Apache Software Foundation. Spamassassin public corpus, 2006. http://spamassassin.apache.org/publiccorpus/.

[6] L. Breiman. Random forests. Mach. Learn. 45(1):5–32, 2001.

[7] M. Chandrasekaran, K. Karayanan, and S. Upadhyaya. Towards phishing e-mail detection based on their structural properties. In New York State Cyber Security Conference, 2006.

[8] N. Chou, R. Ledesma, Y. Teraguchi, and J. C Mitchell. Client-side defense against web-based identity theft. In NDSS, 2004.

[9] W. Cohen. Learning to classify English text with ILP methods. In L. De Raedt, editor, Advances in Inductive Logic Programming, pages 124–143. IOS Press, 1996.

[10] L. Cranor, S. Egelman, J. Hong, and Y. Zhang. Phinding phish: An evaluation of anti-phishing toolbars. Technical report, Carnegie Mellon University, Nov. 2006.

[11] N. Cristianini and J. Shawe-Taylor. An introduction to support Vector Machines: and other kernel-based learning methods. Cambridge University Press, New York, NY, USA, 2000.

[12] FDIC. Putting an end to account-hijacking identity theft, Dec. 2004. http://www.fdic.gov/consumers/consumer/idtheftstudy/identity_theft.pdf.

[13] I. Fette, N. Sadeh, and A. Tomasic. Learning to detect phishing emails. Technical Report CMU-ISRI-06-112, Institute for Software Research, Carnegie Mellon University, June 2006. http://reports-archive.adm.cs.cmu.edu/anon/isri2006/abstracts/06- 112.html.

[14] F. L. Gandon and N. M. Sadeh. Semantic web technologies to reconcile privacy and context awareness. Journal of Web Semantics, 1(3):241–260, 2004.

[15] Gilby Productions. Tinyurl, 2006. http://www.tinyurl.com/.

[16] P. Graham. Better bayesian filtering. In Proceedings of the 2003 Spam Conference, Jan 2003.

[17] B. Leiba and N. Borenstein. A multifaceted approach to spam reduction. In Proceedings of the First Conference on Email and Anti-Spam (CEAS), 2004.

[18] T. Meyer and B. Whateley. Spambayes: Effective open-source, bayesian based, email classification system. In Proceedings of the First Conference on Email and Anti-Spam (CEAS), 2004.

[19] Microsoft. Sender ID framework, 2006. http://www.microsoft.com/senderid.

[20] M. H. Rachna Dhamija, Doug Tygar. Why phishing works. In CHI '06: Proceedings of the SIGCHI conference on Human Factors in computing systems, pages 581–590. ACM Special Interest Group on Computer-Human Interaction, January 2006.